



TUGAS AKHIR - SM141501

**APLIKASI METODE HIDDEN MARKOV
MODEL UNTUK IDENTIFIKASI WAJAH
INDIVIDU**

**ALVINA KHAIRUN NISA'
NRP 1213 100 008**

**Dosen Pembimbing
Drs. Nurul Hidayat, M.Kom**

**DEPARTEMEN MATEMATIKA
Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Teknologi Sepuluh Nopember
Surabaya 2017**



FINAL PROJECT - SM141501

APPLICATION OF HIDDEN MARKOV MODEL METHODS FOR FACE IDENTIFICATION

ALVINA KHAIRUN NISA'
NRP 1213 100 008

Supervisor
Drs. Nurul Hidayat, M.Kom

DEPARTMEN OF MATHEMATICS
Faculty of Mathematics and Natural Science
Sepuluh Nopember Institute of Technology
Surabaya 2017

LEMBAR PENGESAHAN

APLIKASI METODE HIDDEN MARKOV MODEL UNTUK IDENTIFIKASI WAJAH INDIVIDU

APPLICATION OF HIDDEN MARKOV MODEL METHODS FOR FACE IDENTIFICATION


TUGAS AKHIR

Diajukan untuk memenuhi salah satu syarat
Untuk memperoleh gelar Sarjana Sains
Pada bidang studi Ilmu Komputer
Program Studi S-1 Departemen Matematika
Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Teknologi Sepuluh Nopember Surabaya

Oleh :
ALVINA KHAIRUN NISA'
NRP. 1213 100 008


Menyetujui,

Dosen Pembimbing,


Drs. Nurul Hidayat, M.Kom
NIP.19630404 198903 1 002

Mengetahui,

Kepala Departemen Matematika
FMIPA ITS


Dr. Imam Mukhlash, S.Si, MT
NIP. 9700831 199403 1 003
Surabaya, Juli 2017

APLIKASI METODE HIDDEN MARKOV MODEL UNTUK IDENTIFIKASI WAJAH INDIVIDU

Nama : Alvina Khairun Nisa'
NRP : 1213 100 008
Departemen : Matematika
Dosen Pembimbing : Drs. Nurul Hidayat, M.Kom

ABSTRAK

Biometrika merupakan ilmu yang dapat digunakan untuk keperluan identifikasi individu melalui ciri khas yang dimiliki, seperti sidik jari, iris mata, telapak tangan, suara ucapan maupun wajah. Salah satu contoh aplikasi biometrika adalah pada pengenalan individu melalui identifikasi wajah. Pada penelitian ini dilakukan pengenalan individu dengan mengaplikasikan metode *Hidden Markov Model* (HMM) untuk identifikasi wajah. Tahapan dari penelitian ini adalah pengumpulan data citra wajah, *preprocessing*, ekstraksi fitur, *training data*, dan *testing data*. Data *training* terdiri dari 40 individu dengan masing-masing individu mempunyai 4 citra wajah. Sedangkan data *testing* terdiri dari 80 citra wajah individu yang tidak bergerak dan juga dari 40 citra wajah individu yang bergerak. Hasil rata-rata akurasi tertinggi dari penelitian ini adalah 85,21% dengan ukuran citra 64x64 piksel dan barisan observasi $T = 59$. Akurasi pengenalan untuk data *training* adalah 98,13%, 85% untuk akurasi citra wajah individu yang tidak bergerak dan 72,5% untuk akurasi citra wajah individu yang bergerak.

Kata kunci: *Biometrika, Deteksi Wajah, Hidden Markov Model*

APPLICATION OF HIDDEN MARKOV MODEL METHODS FOR FACE IDENTIFICATION

Name : Alvina Khairun Nisa'
NRP : 1213 100 008
Department : Mathematics
Supervisor : Drs. Nurul Hidayat, M.Kom

ABSTRACT

Biometrics is a science that can be used for the purposes of individual identification through their own characteristics, such as fingerprints, irises, palms, sounds or faces. One example of biometric application is the individual recognition through the individual faces identification. In this research, the introduction of individuals by applying the Hidden Markov Model (HMM) for face identification. The process of this research is faces data collection, preprocessing, feature extraction, training data, and testing data. The training data consists of 40 individuals with each individual having 4 face images. While the data testing consist of 80 non-moving individual face images and 40 moving faces image. The highest average accuracy of this research is 85,21% with 64x64 pixel and observation sequence $T = 59$. Accuracy for training data is 98,13%, for non-moving face images testing data is 85% and for moving face images testing data is 72,5%.

**Keywords : Biometric, Face Detection, Hidden Markov
Model**

KATA PENGANTAR

Assalamu'alaikum Wr. Wb.

Alhamdulillahirobbil'aalamiin, segala puji dan syukur bagi Allah SWT yang memiliki apa yang ada di langit dan di bumi dan yang telah memberikan limpahan rahmat, petunjuk serta hidayah-Nya, sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul

“APLIKASI METODE HIDDEN MARKOV MODEL UNTUK IDENTIFIKASI WAJAH INDIVIDU”

sebagai salah satu syarat kelulusan Program Sarjana Departemen Matematika FMIPA Institut Teknologi Sepuluh Nopember (ITS) Surabaya.

Tugas Akhir ini dapat terselesaikan dengan baik berkat bantuan dari berbagai pihak. Suatu kebahagiaan dan kewajiban bagi penulis untuk menyampaikan terima kasih kepada semua pihak yang telah membantu dan mendukung atas terselesainya Tugas Akhir :

1. Ayah, Ibu, Bagus, Dimas serta keluarga besar penulis atas doa dan dukungan yang selalu diberikan kepada penulis
2. Bapak Dr. Imam Mukhlash, S.Si, MT selaku kepala Departemen Matematika Institut Teknologi Sepuluh Nopember Surabaya.
3. Bapak Dr. Didik Khusnul Arif, S.Si., M.Si. selaku Ketua Program Studi S1 Departemen Matematika ITS.

4. Bapak Drs. Iis Herisman, M.Sc selaku Sekretaris Program Studi S1 Departemen Matematika ITS.
5. Ibu Dra. Sri Suprpti H., M.Si selaku dosen wali yang telah memberikan nasihat dan arahan selama penulis menempuh perkuliahan di Departemen Matematika ITS
6. Bapak Drs. Nurul Hidayat, M.Kom selaku dosen pembimbing atas segala arahan, dukungan, dan motivasinya kepada penulis, sehingga Tugas Akhir ini dapat terselesaikan
7. Bapak Dr. Budi Setiyono, S.Si, MT, Bapak Prof. DR. Mohammad Isa Irawan, MT, Ibu Dra. Wahyu Fistia Doctorina, M.Si, dan Ibu Dian Winda S., S.Si. M.Si selaku dosen penguji atas segala saran yang telah diberikan.
8. Bapak dan Ibu dosen, seluruh staf Tata Usaha, dan asisten laboratorium Departemen Matematika ITS
9. Daya, teman penulis yang selalu mau mendukung , memberikan semangat serta tempat bercerita bagi penulis
10. Para sahabat penulis, Nastitie, Frikha, Mimi, Tara, Putri, Upika, Batsa, Fauzia yang selalu memberikan semangat dan dukungan kepada penulis dalam penyelesaian Tugas Akhir ini
11. Teman-teman Matematika ITS 2013 dan SURVIVOR 17 yang selalu memberikan dukungan dan doa kepada penulis
12. Keluarga HIMATIKA ITS khususnya Departemen Hubungan Luar periode 2014/2015,

External Affair periode 2015/2016 atas kerjasamanya untuk membangun Himatika ITS

13. Seluruh pihak yang telah memberikan saran, dukungan, dan motivasi dalam menyelesaikan Tugas Akhir ini

Penulis menyadari bahwa dalam Tugas Akhir ini masih terdapat kekurangan. Oleh sebab itu, kritik dan saran yang bersifat membangun sangat diharapkan oleh penulis. Akhirnya penulis berharap semoga Tugas Akhir ini dapat bermanfaat bagi banyak pihak.

Wassalamu'alaikum Wr. Wb

Surabaya, Juli 2017

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN	v
ABSTRAK	vii
<i>ABSTRACT</i>	ix
KATA PENGANTAR	xi
DAFTAR ISI	xv
DAFTAR GAMBAR	xix
DAFTAR TABEL	xxi
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan	3
1.5 Manfaat	3
1.6 Sistematika Penulisan	3
BAB II TINJAUAN PUSTAKA	5
2.1 Penelitian Terdahulu	5
2.2 Citra Digital	7
2.3 Biometrika	9
2.4 Pengenalan Wajah	11
2.5 Pengolahan Citra	12
2.5.1. Citra Grayscale	12
2.5.2. Deteksi Wajah Viola-Jones	12

2.5.3. <i>Minimum Order-Statistic Filters</i>	14
2.6 <i>Singular Value Decomposition</i>	15
2.7 Metode <i>Hidden Markov Model</i>	16
BAB III METODE PENELITIAN.....	19
3.1 Pengumpulan Data.....	19
3.2 Data Preprocessing	19
3.3 Ekstraksi Blok	19
3.4 Ekstraksi Fitur	19
3.5 Training	20
3.6 Testing.....	20
3.7 Perancangan dan Implementasi Perangkat Lunak.	20
3.8 Uji Coba Sistem	20
BAB IV PERANCANGAN SISTEM.....	23
4.1 Pengambilan Data.....	23
4.2 Data <i>Preprocessing</i>	24
4.3 Ekstraksi Blok	25
4.4 Ekstraksi Fitur	27
4.5 Proses <i>Training</i>	33
4.6 Proses <i>Testing</i>	42
BAB V IMPLEMENTASI SISTEM.....	43
5.1 Lingkungan <i>Hardware</i> dan <i>Software</i>	43
5.2 Implementasi <i>User Interface</i>	43
5.3 Implementasi Praproses Data.....	48

5.4 Implementasi Ekstraksi Fitur	51
5.5 Implementasi proses <i>Training</i>	51
5.6 Implementasi Proses <i>Testing</i>	52
BAB VI UJI COBA DAN EVALUASI SISTEM.....	55
6.1 Dataset Uji Coba <i>Training</i>	55
6.2 Dataset Uji Coba Proses <i>Testing</i>	56
6.3 Hasil Uji Coba Pengenalan.....	56
6.3.1. Uji Coba 1.....	56
6.3.2. Uji Coba 2.....	57
6.3.3. Uji Coba 3.....	57
6.3.4. Uji Coba 4.....	58
6.3.5. Uji Coba 5.....	59
6.4 Pembahasan Hasil Pengujian	60
6.5 Pembahasan Penyebab Besar Kecil Akurasi	61
BAB VII KESIMPULAN DAN SARAN	65
7.1 Kesimpulan	65
7.2 Saran.....	65
DAFTAR PUSTAKA.....	67
LAMPIRAN 1.....	69
LAMPIRAN 2.....	75
LAMPIRAN 3.....	83
BIODATA PENULIS.....	101

DAFTAR GAMBAR

Gambar 2.1. Proses Sampling dan Kuantisasi Citra.	7
Gambar 2.2. Representasi Citra Digital dalam Koordinat (x,y).....	8
Gambar 2.3. Blok Diagram Proses Pengenalan Wajah...	12
Gambar 2.4. Diagram alir metode Viola-Jones.....	13
Gambar 2.5. Jenis Fitur Gambar	14
Gambar 3.1. Blok Diagram Tahapan Penelitian	21
Gambar 4.1. Data <i>Training</i> Afif.....	23
Gambar 4.2. Data <i>Testing</i> Afif.....	24
Gambar 4.3. Barisan <i>Overlapping Blocks</i>	26
Gambar 4.4. Diagram Proses <i>Training</i>	34
Gambar 4.5. Diagram Proses <i>Testing</i>	42
Gambar 5.1. Halaman Awal	44
Gambar 5.2. Halaman <i>Training</i>	45
Gambar 5.3. Halaman <i>Testing</i>	46
Gambar 5.4. Halaman <i>Testing Video</i>	47
Gambar 5.5. Hasil <i>Running Program</i>	48
Gambar 5.6. Hasil <i>Running Program</i>	48
Gambar 5.7. Gambar wajah yang akan di proses	49
Gambar 5.8. Gambar setelah Proses <i>Cropping</i>	50
Gambar 5.9. Setelah Proses <i>Grayscale</i> (a) dan setelah Proses <i>filtering</i> (b).....	50
Gambar 6.1. Data <i>Training</i> Afif.....	62
Gambar 6.2. Data <i>Testing</i> Afif (tidak bergerak)	62
Gambar 6.3. Data <i>Testing</i> Afif (bergerak).....	63

DAFTAR TABEL

Tabel 1. Kegunaan Menu-Menu dalam <i>Menu Bar</i>	44
Tabel 2. Kegunaan Tombol dalam Halaman <i>Testing</i>	46
Tabel 3. Data <i>Training</i>	55
Tabel 4. Hasil Uji Coba 1 dengan Data <i>Training</i>	56
Tabel 5. Hasil Uji Coba 1 dengan Data <i>Testing</i>	56
Tabel 6. Hasil Uji Coba 2 dengan Data <i>Training</i>	57
Tabel 7. Hasil Uji Coba 2 dengan Data <i>Testing</i>	57
Tabel 8. Hasil Uji Coba 3 dengan Data <i>Training</i>	58
Tabel 9. Hasil Uji Coba 3 dengan Data <i>Testing</i>	58
Tabel 10. Hasil Uji Coba 4 dengan Data <i>Training</i>	58
Tabel 11. Hasil Uji Coba 4 dengan Data <i>Testing</i>	59
Tabel 12. Hasil Uji Coba 5 dengan Data <i>Training</i>	59
Tabel 13. Hasil Uji Coba 5 dengan Data <i>Testing</i>	59
Tabel 14. Akurasi Sistem secara Keseluruhan.....	61

BAB I

PENDAHULUAN

Pada bab pendahuluan akan dijelaskan mengenai hal-hal yang melatarbelakangi permasalahan pada Tugas Akhir ini. Kemudian, dijabarkan dalam rumusan masalah, batasan masalah, tujuan, dan manfaat yang bisa diambil dari penyusunan Tugas Akhir ini.

1.1 Latar Belakang

Dalam perkembangan teknologi informasi dan komunikasi (TIK), identifikasi merupakan bagian penting dalam terjaminnya kerahasiaan personal/data. Biometrika, yang merupakan cabang matematika terapan yang muncul dan berkembang pada era globalisasi, merupakan ilmu yang digunakan untuk keperluan identifikasi individu melalui sidik jari, retina mata, telapak tangan, suara maupun wajah. Biometrika sendiri bukan merupakan teknologi baru karena sebelum biometrika diterapkan secara digital, biometrika sudah diterapkan secara manual contohnya melalui tanda tangan dengan cap sidik jari. Setelah biometrika dikembangkan dengan menggunakan komputerisasi, biometrika mengalami perkembangan yang pesat[1].

Biometrika dapat diaplikasikan pada pembuatan teknologi yang berbasis pada identifikasi wajah individu, seperti sistem pengenalan individu. Hal ini sangat dimungkinkan karena bidang garap biometrika adalah mengidentifikasi individu berdasarkan ciri khas biologis yang dimiliki oleh individu tersebut, antara lain adalah wajah. Wajah merupakan bagian individu yang mudah dan alami yang dapat digunakan untuk mengenali setiap individu. Banyak aplikasi yang menerapkan sistem pengenalan melalui wajah. Antara lain dalam sistem keamanan seperti ijin akses masuk ruangan, pengawasan

lokasi, maupun dalam pencarian identitas individu. Penelitian terhadap pengenalan wajah sudah banyak sekali dilakukan dengan kelebihan dan kekurangan tertentu. Hal ini karena wajah manusia mempresentasikan sesuatu yang kompleks, sehingga kerumitan untuk mengembangkan model komputasi yang ideal untuk pengenalan wajah manusia tidak jarang dialami.

Sistem pengenalan wajah telah banyak diaplikasikan dengan menggunakan berbagai metode. Salah satunya menggunakan metode *Hidden Markov Model* (HMM). HMM cukup populer digunakan dalam berbagai bidang kecerdasan buatan dan bioinformatika. Hal ini diakibatkan oleh dua alasan kuat. Pertama, model ini sangat kaya dalam struktur matematika sehingga dapat digunakan sebagai teori dasar di banyak bidang aplikasi. Kedua, jika model ini digunakan dengan benar, ia dapat bekerja dengan efektif dan efisien untuk implementasi beberapa aplikasi penting [2]. Oleh karena itu, dalam penelitian tugas akhir ini dibuat sebuah sistem, dalam bentuk program/perangkat lunak, untuk mengenali individu melalui identifikasi wajah individu menggunakan metode *Hidden Markov Model*.

1.2 Rumusan Masalah

Berdasarkan uraian latar belakang, rumusan masalah yang akan diselesaikan pada Tugas Akhir ini adalah :

1. Mengaplikasikan metode *Hidden Markov Model* dalam mengidentifikasi wajah individu.
2. Membuat program/perangkat lunak untuk mengenali individu melalui identifikasi wajah.

1.3 Batasan Masalah

Batasan dalam permasalahan di atas adalah :

1. Ekspresi wajah yang dikenali dibatasi pada ekspresi wajah normal.

2. Software yang digunakan untuk membuat aplikasi pengenalan wajah adalah MATLAB.
3. Individu yang akan dikenali tidak mengenakan aksesoris yang menutupi wajah.

1.4 Tujuan

Tujuan umum dari penelitian ini adalah untuk mendapatkan program/perangkat lunak yang dapat digunakan untuk mengenali individu/orang.

Sedangkan tujuan khusus untuk tugas akhir ini adalah sebagai berikut:

1. Melakukan identifikasi wajah individu dengan mengaplikasikan metode *Hidden Markov Model*.
2. Melakukan ekstraksi fitur wajah individu dengan mengaplikasikan algoritma *Singular Value Decomposition*.

1.5 Manfaat

Manfaat yang diharapkan dari tugas akhir ini adalah:

1. Sistem deteksi dini terhadap orang yang memasuki area yang membutuhkan keamanan tinggi.
2. Membuat pintu elektronik berbasis identifikasi wajah.

1.6 Sistematika Penulisan

Penulisan Tugas Akhir ini disusun dalam lima bab, yaitu:

1. BAB I PENDAHULUAN

Bab ini berisi tentang gambaran umum dari penulisan Tugas Akhir yang meliputi latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat, dan sistematika penulisan.

2. **BAB II TINJAUAN PUSTAKA**

Bab ini menjelaskan tentang studi penelitian sebelumnya dan beberapa teori pendukung yang berkaitan dengan Tugas Akhir ini.

3. **BAB III METODE PENELITIAN**

Bab ini menjelaskan tahapan-tahapan dan metode yang digunakan untuk menyelesaikan Tugas Akhir ini.

4. **BAB IV PERANCANGAN SISTEM**

Bab ini menjelaskan tahap persiapan pengolahan data hingga proses konstruksi sistem sebagai acuan dalam implementasi sistem.

5. **BAB V IMPLEMENTASI SISTEM**

Bab ini membahas proses implementasi dengan menggunakan MATLAB berdasarkan rancangan sistem yang telah dibuat pada bab sebelumnya.

6. **BAB VI UJI COBA DAN EVALUASI SISTEM**

Bab ini membahas tahap-tahap uji coba berdasarkan implementasi sistem yang telah dibuat beserta pengujian kinerja sistem.

7. **BAB VII KESIMPULAN DAN SARAN**

Bab ini berisi kesimpulan Tugas Akhir yang diperoleh dari bab uji coba dan evaluasi sistem serta saran untuk pengembangan penelitian selanjutnya

BAB II

TINJAUAN PUSTAKA

Pada bab ini diuraikan mengenai penelitian terdahulu, dan dasar teori yang digunakan dalam penyusunan tugas akhir.

2.1 Penelitian Terdahulu

Sistem pengenalan wajah manusia merupakan salah satu bidang yang cukup berkembang dewasa ini, dimana aplikasi dapat diterapkan dalam bidang keamanan (*security system*) seperti ijin akses masuk ruangan, pengawasan lokasi (*surveillance*), maupun pencarian identitas individu pada database kepolisian.

Salah satu penelitian tentang sistem pengenalan wajah adalah tugas akhir dengan judul “*Convolutional Neural Networks* untuk Pengenalan Wajah Secara *Real-Time*” yang dilakukan oleh Muhammad Zufar pada tahun 2016. Dalam penelitian ini menjelaskan bahwa pembuatan sistem pengenalan wajah dilakukan dengan menggunakan salah satu metode *deep neural networks* yaitu *Convolutional Neural Networks* (*CNN*) sebagai pengenalan wajah secara real-time. Dalam penyusunan arsitektur *CNN* dilakukan konfigurasi inisialisasi parameter untuk mempercepat proses training jaringan. Hasil uji coba dengan menggunakan konstruksi model *CNN* sampai kedalaman 7 lapisan dengan input dari hasil ekstraksi *Extended Local Binary Pattern* dengan radius 1 dan neighbor 15 menunjukkan kinerja pengenalan wajah meraih rata-rata tingkat akurasi lebih dari 89% dalam 2 frame per detik[3]. Akan tetapi dalam penelitian tersebut menggunakan Web Cam M-Tech 5MP untuk menangkap citra wajah yang akan dikenali dan juga menggunakan Metode *CNN*.

Pada penelitian lain yang telah menerapkan metode HMM adalah tugas akhir dengan judul “Prediksi Sekuens DNA Berdasarkan Data Ekspresi Gen Menggunakan Model Hidden Markov ” oleh Harmuda Pandiangan tahun 2016. Sekuens DNA ini diprediksi berdasarkan data ekspresi gen. Dalam

penelitiannya, Pandiangan menggunakan model Hidden markov dengan algoritma Viterbi, asam amino sebagai *observable state* dan rangkaian kodon sebagai *hidden state*. [4]

Penelitian lainnya yang pernah pula menggunakan metode HMM yaitu tugas akhir dengan judul "Sistem Pengenalan Wajah (*Face Recognition*) Menggunakan Metode Hidden Markov Model (HMM)" yang dilakukan oleh Sepritahara pada tahun 2012. Sistem pengenalan wajah manusia menggunakan metode Hidden Markov Models (HMM) mencapai tingkat akurasi pengenalan sebesar 84,28%, dengan database 70 gambar yang terdiri dari 10 individu dengan masing-masing individu memiliki 7 variasi ekspresi yang berbeda [5]. Namun dalam penelitian tersebut masukan citra yang akan dikenali berupa citra gambar yang hanya menampilkan bagian wajah individunya saja.

Penelitian lainnya yang juga menggunakan metode HMM yaitu "*A Hidden Markov Model-Based Isolated and Meaningful Hand Gesture Recognition*" yang dilakukan oleh Mahmoud Elmezain, Ayoub Al-Hamadi, Jorg Appenrodt, dan Bernd Michaelis pada tahun 2009. Pada penelitian ini mengusulkan sebuah sistem untuk mengenali gerakan satu tangan yang menggambarkan angka secara real-time menggunakan HMM. Database berupa 30 video secara terpisah gerakan angka 0 sampai dengan 9 dan juga 70 video secara langsung dari angka 0 sampai 9. Hasil yang diberikan adalah untuk rata-rata pengenalan gerak keseluruhan mencapai 98.6% dan 94.2% untuk masing-masing gerakan. Tetapi dalam penelitian tersebut menggunakan gerakan tangan untuk mengenali angka yang digambarkan [6].

Penelitian lainnya yang menggunakan metode HMM adalah "*A New Fast and Efficient HMM-Based Face Recognition System Using a 7-State HMM Along With SVD Coefficients*" oleh H. Miar-Naimi dan P. Davari pada tahun 2008. Pada penelitian ini sistem pengenalan individu dalam identifikasi wajah menggunakan 7 state *Hidden Markov Model* dan SVD koefisien. Hasil yang akurasi diperoleh adalah 100% tetapi data yang digunakan adalah *face database* dari Olivetti

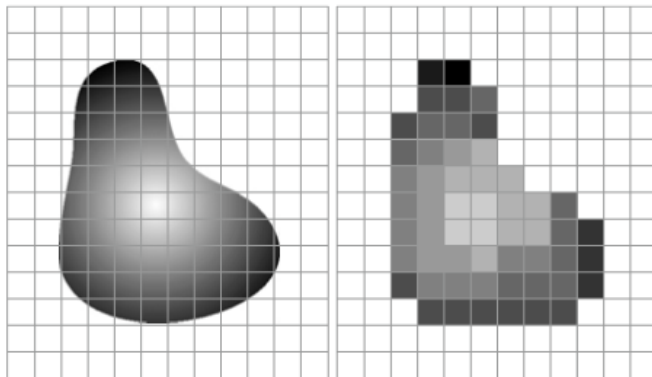
Research Laboratory (ORL) dan *YALE* database.[12] Tetapi pada penelitian ini database yang digunakan adalah data sekunder.

Oleh karena itu, pada penelitian tugas akhir ini dibuat program/perangkat lunak pengenalan wajah individu menggunakan metode *Hidden Markov Model*, data yang digunakan adalah data primer.

2.2 Citra Digital

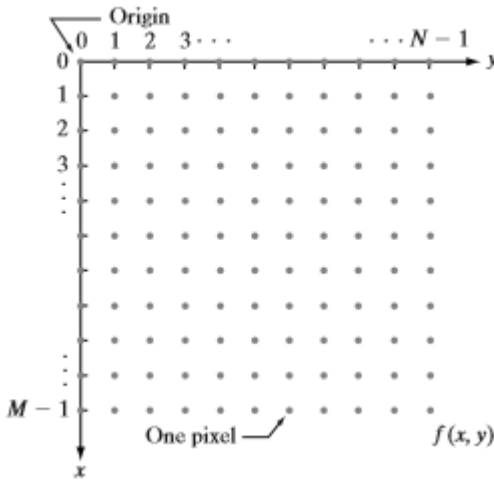
Citra adalah representasi, kemiripan atau imitasi dari suatu objek atau benda[7]. Citra terbagi 2 yaitu ada citra yang bersifat analog dan ada citra yang bersifat digital. Citra analog adalah citra yang bersifat kontinu seperti gambar pada monitor televisi, foto sinar X, hasil CT Scan dan lain-lain. Sedangkan pada citra digital adalah citra yang dapat diolah oleh komputer[8].

Suatu citra agar dapat direpresentasikan secara numerik, maka citra harus didigitalisasi, baik terhadap ruang (koordinat (x,y)) maupun terhadap skala keabuannya ($f(x,y)$). Proses digitalisasi koordinat (x,y) dikenal sebagai pencuplikan citra (*image sampling*), sedangkan proses digitalisasi skala keabuan ($f(x,y)$) disebut sebagai kuantisasi derajat keabuan (*gray level quantization*). Proses sampling dan kuantisasi dapat dilihat pada Gambar 2.1.



Gambar 2.1. Proses Sampling dan Kuantisasi Citra.

Misalkan sebuah citra $f(x,y)$ dicuplik sehingga menghasilkan citra digital dengan M baris dan N kolom. Nilai dari koordinat (x,y) sekarang menjadi kuantitas diskrit. Dengan demikian, nilai koordinat pada titik asal adalah $(x,y) = (0,0)$. Nilai koordinat berikutnya sepanjang baris pertama dari citra direpresentasikan sebagai $(x,y) = (0,1)$. Notasi $(0,1)$ digunakan untuk menandakan sampel kedua sepanjang baris pertama. Representasi citra digital dalam koordinat (x,y) dapat dilihat pada Gambar 2.2.



Gambar 2.2. Representasi Citra Digital dalam Koordinat (x,y)

Sebuah citra digital dapat mewakili oleh sebuah matriks yang terdiri dari M kolom N baris, dimana perpotongan antara kolom dan baris disebut piksel (piksel = picture element), yaitu elemen terkecil dari sebuah citra. Piksel mempunyai dua parameter, yaitu koordinat dan intensitas atau warna. Untuk sebuah citra digital, setiap piksel memiliki nilai integer yakni *gray level* yang menunjukkan amplitudo atau intensitas dari piksel tersebut. Citra merupakan fungsi dua dimensi yang kedua variabelnya yaitu nilai amplitudo dan koordinatnya merupakan

nilai integer. Oleh sebab itu, sebuah citra digital dapat ditulis dalam bentuk matriks berikut.

$$f(x, y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,M-1) \\ f(1,0) & \dots & \dots & f(1,M-1) \\ \dots & \dots & \dots & \dots \\ f(N-1,0) & f(N-1,1) & \dots & f(N-1,M-1) \end{bmatrix} \quad (2.1)$$

Berdasarkan gambaran tersebut, secara matematis citra digital dapat dituliskan sebagai fungsi intensitas $f(x,y)$, dimana harga x (baris) dan y (kolom) merupakan koordinat posisi dan $f(x,y)$ adalah nilai fungsi pada setiap titik (x,y) yang menyatakan besar intensitas citra atau tingkat keabuan atau warna dari piksel di titik tersebut. Pada proses digitalisasi (sampling dan kuantitas) diperoleh besar baris M dan kolom N hingga citra membentuk matriks $M \times N$ dan jumlah tingkat keabuan piksel G [7].

Pada beberapa diskusi, untuk menunjukkan citra digital dan elemen-elemennya menggunakan matriks yang lebih tradisional yaitu:

$$A = \begin{bmatrix} a_{0,0} & a_{0,1} & \dots & a_{0,N-1} \\ a_{1,0} & a_{1,1} & \dots & a_{1,N-1} \\ \dots & \dots & \dots & \dots \\ a_{M-1,0} & a_{M-1,1} & \dots & a_{M-1,N-1} \end{bmatrix} \quad (2.2)$$

Terlihat bahwa $a_{ij} = f(x = i, y = j) = f(i, j)$, sehingga persamaan (2.1) dan (2.2) adalah matrik yang sama.

2.3 Biometrika

Biometrika (berasal dari bahasa Yunani *bios* yang artinya hidup dan *metron* yang artinya mengukur) secara umum adalah studi tentang karakteristik biologi yang terukur. Dalam dunia teknologi informasi, biometrika relevan dengan teknologi yang digunakan untuk menganalisa fisik dan kelakuan manusia dalam autentifikasi.

Sebuah sistem biometrika pada dasarnya sistem pengenalan pola yang mengakui seseorang berdasarkan vektor fitur yang berasal dari karakteristik fisiologis atau perilaku yang dimiliki seseorang. Tergantung pada konteks aplikasi, sistem

biometrika biasanya beroperasi di salah satu dari dua mode: verifikasi atau identifikasi. Dalam modus verifikasi, sistem memvalidasi identitas seseorang dengan membandingkan ciri-ciri biometrika yang sudah disimpan dalam database. Verifikasi identitas biasanya digunakan untuk pengakuan positif, dimana tujuannya adalah untuk mencegah banyak orang dari menggunakan identitas sama. Dalam modus identifikasi, sistem mengakui seorang individu dengan mencari database seluruh template untuk perbandingan. Sistem melakukan perbandingan satu ke banyak data untuk membangun identitas seseorang. Identifikasi biasanya digunakan untuk pengakuan negatif, dimana tujuannya untuk mencegah satu orang menggunakan beberapa identitas. Sementara metode pengakuan tradisional password, pin, tombol, dan token bekerja untuk pengakuan positif, hanya biometrika dapat digunakan untuk pengakuan negatif[1].

Tidak semua bagian tubuh atau perilaku seseorang dapat digunakan sebagai biometrika. Ada beberapa persyaratan yang harus dipenuhi agar bagian-bagian tubuh atau perilaku manusia dapat digunakan sebagai biometrika, antara lain[1] :

1. Universal (*universality*), artinya karakteristik yang dipilih harus dimiliki oleh setiap orang.
2. Membedakan (*distinctiveness*), artinya karakteristik yang dipilih memiliki kemampuan membedakan antara satu orang dengan orang lain.
3. Permanen (*permanence*), artinya karakteristik yang dipilih tidak cepat berubah dalam periode waktu yang lama.
4. Kolektabilitas (*collectability*), artinya karakteristik yang dipilih mudah diperoleh dan dapat diukur secara kuantitatif.
5. Unjuk kerja (*performance*), artinya karakteristik yang dipilih dapat memberikan unjuk kerja yang bagus baik dari segi akurasi maupun kecepatan, termasuk sumber daya yang dibutuhkan untuk memperolehnya.
6. Dapat diterima (*acceptability*), artinya masyarakat mau menerima karakteristik yang digunakan.

7. Tidak mudah dikelabui (*circumvention*), artinya karakteristik yang dipilih tidak mudah dikelabui dengan berbagai cara curang.

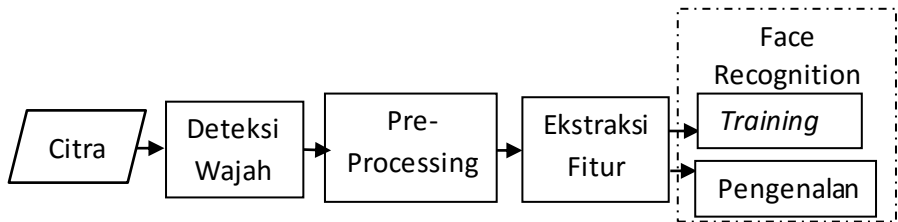
Syarat 1–4 merupakan syarat utama yang harus dipenuhi, sedangkan 5-7 merupakan syarat tambahan. Bila karakteristik yang dipilih memenuhi persyaratan di atas maka karakteristik tersebut dapat digunakan untuk biometrika.

Aplikasi biometrika terbagi menjadi tiga kelompok utama[1]:

- Aplikasi komersial, seperti login komputer jaringan, keamanan data elektronik, e-commerce, akses Internet, ATM, kartu kredit, kontrol akses fisik, HP, PDA, dan catatan manajemen medis.
- Aplikasi pemerintah seperti KTP, SIM, jaminan sosial, dan pemeriksaan paspor.
- Aplikasi forensik seperti identifikasi mayat, penyelidikan pidana, identifikasi teroris, penentuan orangtua dan anak-anak hilang.

2.4 Pengenalan Wajah

Wajah merupakan bagian dari tubuh manusia yang menjadi fokus perhatian di dalam interaksi sosial, wajah memainkan peranan vital dengan menunjukkan identitas dan emosi. Kemampuan manusia untuk mengetahui seseorang dari wajahnya sangat luar biasa. Kita dapat mengenali ribuan wajah karena frekuensi interaksi yang sangat sering ataupun hanya sekilas bahkan dalam rentang waktu yang sangat lama. Bahkan kita mampu mengenali seseorang walaupun terjadi perubahan pada orang tersebut karena bertambahnya usia atau pemakaian kacamata atau perubahan gaya rambut. Oleh karena itu wajah digunakan sebagai organ dari tubuh manusia yang dijadikan indikasi pengenalan seseorang atau *face recognition*[5]. Diagram pengenalan wajah dapat dilihat pada gambar 2.3.



Gambar 2.3. Blok Diagram Proses Pengenalan Wajah

2.5 Pengolahan Citra

Pengolahan citra merupakan pemrosesan citra, khususnya dengan menggunakan komputer untuk menjadikan kualitas citra menjadi lebih baik. Pengolahan citra bertujuan untuk memperbaiki kualitas citra agar mudah diinterpretasi oleh manusia atau komputer[8].

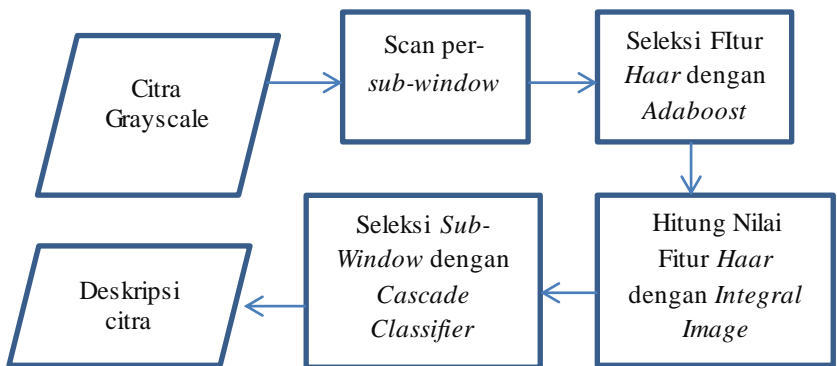
2.5.1. Citra Grayscale

Dalam komputasi, suatu citra digital *grayscale* adalah suatu citra dimana nilai dari setiap piksel merupakan simple tunggal. Citra yang ditampilkan dari citra jenis ini terdiri atas warna abu-abu, bervariasi pada warna hitam pada bagian yang intensitas terlemah dan putih pada intensitas yang terkuat. Citra *grayscale* berbeda dengan citra hitam-putih, dimana pada konteks komputer, citra hitam-putih hanya terdiri dari atas 2 warna yaitu hitam dan putih saja. Citra grayscale disimpan dalam format 8 bit untuk setiap simple pixel, yang memungkinkan sebanyak 256 intensitas. Format ini sangat membantu dalam pemrograman karena manipulasi bit yang tidak terlalu banyak[10].

2.5.2. Deteksi Wajah Viola-Jones

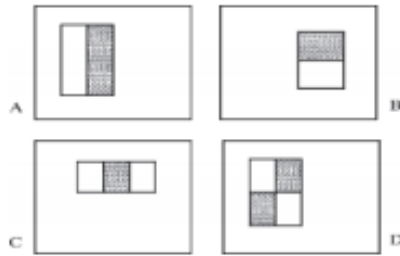
Objek wajah dicari menggunakan *Viola-Jones* yang mengacu alur seperti Gambar 2.4, dimana citra *grayscale* akan di-*scan* per-sub-window untuk dicari fitur positif dengan *Adaboost* dan *Cascade Classifier*. Jika wajah terdeteksi, akan

dilakukan penggambaran garis persegi pada wajah tersebut. Prosedur deteksi wajah Viola-Jones mengklasifikasikan gambar berdasarkan pada nilai fitur sederhana. Terdapat banyak alasan untuk menggunakan fitur daripada piksel secara langsung. Alasan yang paling umum adalah bahwa fitur dapat digunakan untuk mengkodekan pengetahuan domain ad-hoc yang sulit dalam pembelajaran terhadap data latih yang terbatas jumlahnya. Alasan penting kedua untuk menggunakan fitur adalah sistem fitur berbasis operasi jauh lebih cepat daripada sistem berbasis *pixel* [15].



Gambar 2.4. Diagram alir metode Viola-Jones

Klasifikasi gambar dilakukan berdasarkan nilai dari sebuah fitur. Penggunaan fitur dilakukan karena pemrosesan fitur berlangsung lebih cepat dibandingkan pemrosesan citra per-piksel. Terdapat tiga jenis fitur berdasarkan jumlah persegi panjang yang terdapat di dalamnya, seperti yang dapat dilihat pada Gambar 2.5[15] :



Gambar 2.5. Jenis Fitur Gambar

Pada Gambar 3 menggambarkan bahwa fitur (a) dan (b) terdiri dari dua persegi panjang, sedangkan fitur (c) terdiri dari tiga persegi panjang dan fitur (d) empat persegi panjang. Cara menghitung nilai dari fitur ini adalah mengurangi nilai piksel pada area putih dengan piksel pada area hitam.

Di dalam tiap sub-*window image*, jumlah total dari fitur Haar sangat besar, jauh lebih besar jika dibandingkan dengan jumlah *pixel*. Untuk memastikan pengklasifikasian dapat dilakukan secara cepat, proses pembelajaran harus menghilangkan fitur-fitur mayoritas yang tersedia, dan memusatkan pada sekumputa kecil fitur yang perlu. *Adaboost* bertujuan untuk membentuk *template* wajah.

Suatu metode klasifikasi yang menggunakan beberapa tingkatan dalam penyeleksian. Pada tiap tingkatan dilakukan penyeleksian menggunakan algoritma *Adaboost* yang telah di-*training* dengan menggunakan Fitur Haar. Penyeleksian berguna untuk memisahkan antara sub-*window* yang mengandung positif objek dengan negatif objek [15]. Positif objek adalah gambar yang terdeteksi memiliki objek yang diinginkan sedangkan negatif objek, adalah gambar yang terdeteksi tidak memiliki objek yang diinginkan.

2.5.3. Minimum Order-Statistic Filters

Order-statistic filter pada umumnya adalah filter spasial *non-linier* yang hasilnya didasarkan pada urutan (rangking) piksel yang mengisi area citra yang diapit filter dan kemudian mengganti nilai dari pusat pixel 3x3 dengan nilai yang

ditentukan oleh hasil perangkingan. Salah satu *Order-statistic* yang digunakan adalah minimum filter untuk memilih nilai terkecil.[12] Untuk filter ini dapat menggunakan persamaan (2.3)

$$f'(x, y) = \min_{(s,t) \in S_{xy}} \{g(s, t)\} \quad (2.3)$$

Keterangan :

$f'(x, y)$ = Hasil minimum filter

$g(s, t)$ = *grey level* piksel (s,t)

S_{xy} = daerah yang diliputi piksel

2.6 Singular Value Decomposition

Algoritma *Singular Value Decomposition* (SVD) adalah faktorisasi nyata pada matriks kompleks, [13] dimana dalam transformasi SVD, matriks asli dapat diuraikan menjadi tiga matriks dengan ukuran yang sama, namun jika dikalikan ketiga matriks yang telah diuraikan, maka akan sama dengan matriks asli. [14] Algoritma SVD dapat dilihat dalam persamaan (2.4) dan persamaan (2.5).

$$X = USV^T \quad (2.4)$$

$$[X]_{m \times n} = [U]_{m \times n} \cdot [S]_{n \times n} \cdot [V^T]_{n \times n} \quad (2.5)$$

Keterangan:

U : matrik orthogonal berukuran m x n

S : matrik diagonal berukuran n x n

V^T : matrik orthogonal berukuran n x n

Dalam perhitungan SVD pertama-tama kita perlu menghitung nilai eigen dan vektor eigen dari XX^T dan $X^T X$. Vektor eigen dari XX^T bentuk kolom dari U, sedangkan vektor eigen dari $X^T X$ suatu bentuk kolom V. Selain itu, nilai-nilai singular (SV_s) di S adalah akar kuadrat dari nilai eigen dari XX^T atau $X^T X$.

2.7 Metode *Hidden Markov Model*

Metode *Hidden Markov Model* merupakan pemodelan probabilitas suatu sistem dengan mencari parameter-parameter markov yang tidak diketahui untuk memperoleh analisis sistem tersebut. *Metode Hidden Markov Model* (HMM) mampu menangani perubahan statistik dari gambar, dengan memodelkan elemen-elemen menggunakan probabilitas. Salah satu aplikasinya adalah pada *image processing*, HMM memiliki tiga parameter utama yang harus dicari nilainya terlebih dahulu, ketiga parameter tersebut sebagai berikut.

Parameter A yang disebut juga probabilitas transisi state dalam HMM dinyatakan dalam sebuah matriks dengan ukuran $N \times N$ dengan N adalah jumlah state yang ada. Matriks A yang terbentuk ditunjukkan oleh persamaan (2.6).

$$a_{ij} = P[q_{t+1} = S_j | q_t = S_i] \quad 1 \leq i, j \leq N \quad (2.6)$$

Keterangan :

a_{ij} = probabilitas transisi dari state i ke state j

q_t = kondisi saat t

P = peluang/probabilitas

S_j = state ke j

q_{t+1} = kondisi sesudah q_t

S_i = state ke i

N = jumlah state

Parameter B disebut sebagai probabilitas observasi, Matriks B yang terbentuk ditunjukkan oleh persamaan (2.7)

$$b_j(k) = P[v_k \text{ terhadap } t | q_t = S_j] \quad 1 \leq j \leq N, 1 \leq k \leq M \quad (2.7)$$

Keterangan :

$b_j(k)$ = probabilitas distribusi matriks observasi

P = peluang/probabilitas

v_k = probabilitas simbol pengamatan pada state j

q_t = kondisi saat t
 S_j = state ke j

Parameter π , disebut sebagai parameter awal, merupakan probabilitas kemunculan suatu state di awal. Parameter π ditunjukkan dalam persamaan 2.8

$$\pi = \{\pi_1, \pi_2, \dots, \pi_N\} \quad (2.8)$$

Hidden markov model dapat dituliskan sebagai $\lambda = (A, B, \pi)$.

BAB III

METODE PENELITIAN

Berikut adalah tahapan-tahapan yang harus dilakukan dalam penelitian Tugas Akhir ini :

3.1 Pengumpulan Data

Pengumpulan data dilakukan untuk memperoleh informasi yang dibutuhkan untuk mencapai tujuan Tugas Akhir. Data yang digunakan dalam Tugas Akhir ini berupa citra wajah individu. Proses pengumpulan data dilakukan secara manual dengan menggunakan kamera digital.

3.2 Data Preprocessing

Proses yang dilakukan untuk memperbaiki citra inputan dengan melakukan proses deteksi wajah, dalam penelitian ini menggunakan algoritma standar Paul Viola dan Michael Jones dalam jurnalnya "*Rapid Object Detection using a Boosted Cascade of Simple Features*" tahun 2001 yang sudah terdapat dalam MATLAB.[15] Setelah itu melakukan desaturasi, yaitu proses mengubah citra berwarna tiga channels (*red, green, blue layer*) menjadi hanya satu channel yaitu citra keabuan (*grayscale*). Kemudian dilakukan filtering menggunakan *minimum order-statistic filtering*.

3.3 Ekstraksi Blok

Proses yang dilakukan untuk mengubah 2D matrik dari citra ke dalam bentuk 1D barisan observasi.

3.4 Ekstraksi Fitur

Dalam penelitian ini ekstraksi fitur akan menggunakan algoritma *Singular Value Decomposition* (SVD).

3.5 Training

Setelah proses ekstraksi fitur akan didapatkan *sequence* untuk pembentukan parameter HMM pada tiap individu.

3.6 Testing

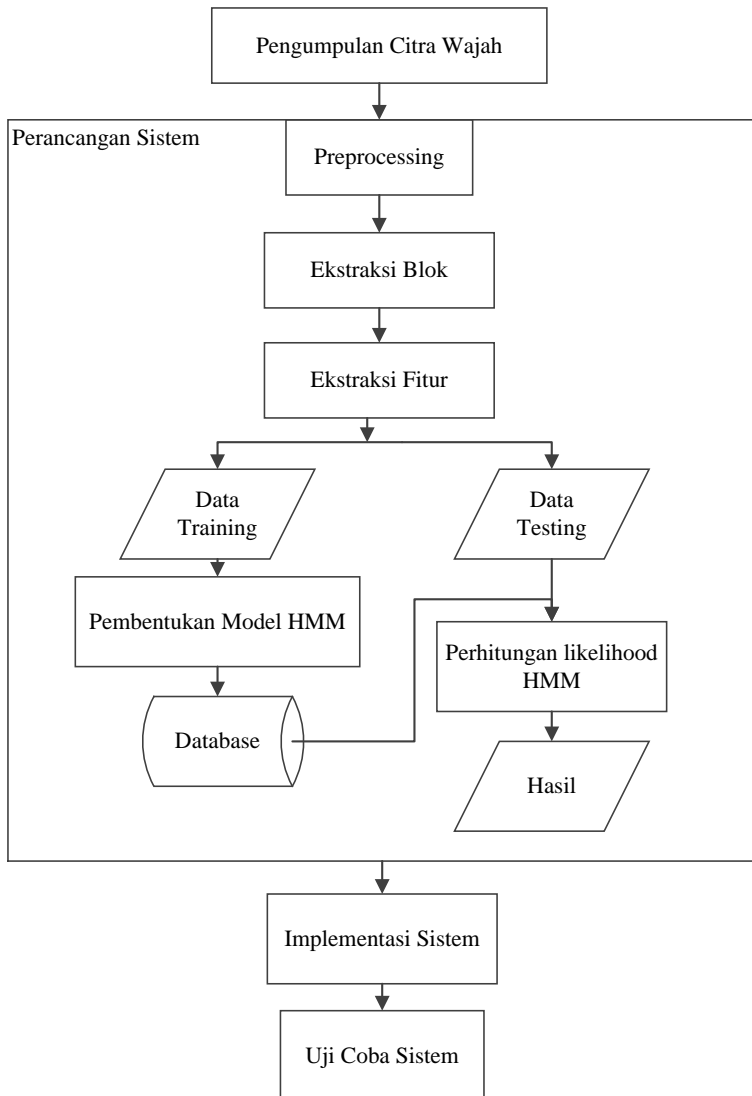
Pada proses *testing* dilakukan evaluasi model dengan cara sistem akan membandingkan model yang terdapat pada database dengan citra yang diuji.

3.7 Perancangan dan Implementasi Perangkat Lunak

Pada tahap ini dibuat perancangan *interface* yaitu membuat visualisasi dari sistem pengenalan wajah individu agar mudah dipahami oleh pembaca. Implementasi perangkat lunak dikerjakan dengan menggunakan software MATLAB.

3.8 Uji Coba Sistem

Setelah melakukan implementasi, selanjutnya akan dilakukan pengujian untuk memeriksa apakah hasilnya sudah sesuai atau terjadi error. Jika masih terjadi error maka akan dilakukan evaluasi dan perbaikan pada program sehingga didapatkan program yang sesuai dengan keinginan.



Gambar 3.1. Blok Diagram Tahapan Penelitian

BAB IV

PERANCANGAN SISTEM

Bab ini menjelaskan rancangan desain sistem yang digunakan sebagai acuan untuk implementasi sistem. Desain sistem menggambarkan proses rancang bangun secara terperinci dari awal tahap pengumpulan data hingga proses konstruksi sistem.

4.1 Pengambilan Data

Data yang dikumpulkan adalah data gambar individu untuk data *training* dan data *testing* dengan rincian citra wajah bergerak dan citra wajah yang tidak bergerak. Data gambar *training* adalah gambar yang berwarna dengan ukuran yang bervariasi yang terdiri dari 40 individu dengan masing-masing individu mempunyai 4 gambar seperti yang terlihat pada Gambar 4.1 dan untuk data *testing* terdiri dari 120 citra wajah individu yang dibedakan menjadi 2 yaitu citra wajah bergerak yang terdiri dari 40 citra wajah yang diambil dari frame video dan citra wajah yang tidak bergerak yang terdiri dari 80 citra wajah seperti yang terlihat pada Gambar 4.2. Data *training* selengkapnya dapat dilihat pada LAMPIRAN 1.



Gambar 4.1. Data Training Afif



Gambar 4.2. Data Testing Afif

4.2 Data Preprocessing

Praproses ini terdiri dari proses deteksi wajah, *cropping*, konversi citra warna ke *grayscale*, dan penyamaan ukuran citra. Algoritma deteksi wajah yang digunakan adalah algoritma standar Paul Viola dan Michael Jones. Setelah dilakukan deteksi wajah kemudian wajah yang terdeteksi dilakukan proses *cropping*. Selanjutnya citra berwarna yang telah melalui proses *cropping* akan dikonversi menjadi citra *grayscale* dan dilakukan penyamaan ukuran citra. Berikut merupakan contoh matrik dari citra *grayscale* dengan ukuran 8 x 8

$$I_m = \begin{bmatrix} 189 & 184 & 146 & 112 & 78 & 64 & 57 & 57 \\ 194 & 158 & 81 & 63 & 59 & 56 & 56 & 58 \\ 179 & 99 & 80 & 72 & 57 & 57 & 54 & 55 \\ 150 & 87 & 86 & 62 & 55 & 56 & 55 & 54 \\ 111 & 68 & 58 & 56 & 54 & 54 & 54 & 53 \\ 75 & 61 & 57 & 56 & 55 & 54 & 54 & 53 \\ 67 & 59 & 55 & 52 & 52 & 52 & 52 & 51 \\ 59 & 58 & 57 & 52 & 52 & 52 & 51 & 51 \end{bmatrix}$$

Kemudian dilakukan *filtering* dengan menggunakan *minimum order-statistic filtering* sesuai persamaan(2.2) untuk mengurangi *noise* dan mengurangi pantulan cahaya yang terdapat di mata. Sehingga matrik I_m menjadi :

$$I = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 80 & 63 & 57 & 56 & 54 & 54 & 0 \\ 0 & 80 & 62 & 55 & 55 & 54 & 54 & 0 \\ 0 & 58 & 56 & 54 & 54 & 54 & 53 & 0 \\ 0 & 57 & 56 & 54 & 54 & 54 & 53 & 0 \\ 0 & 55 & 52 & 52 & 52 & 52 & 51 & 0 \\ 0 & 55 & 52 & 52 & 52 & 51 & 51 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

4.3 Ekstraksi Blok

Setelah dilakukan *preprocessing*, kemudian dilakukan SVD untuk menentukan barisan observasinya. Sebuah gambar mempunyai ukuran 2D matrik sehingga gambar tersebut dirubah menjadi 1D barisan observasi untuk mempermudah perhitungan dalam proses selanjutnya. Masing-masing gambar tersebut mempunyai panjang W dan lebar H. Gambar tersebut dibagi menjadi *overlapping blocks* dengan panjang L dan lebar W. Ilustrasi ekstraksi blok dapat dilihat dalam Gambar 4.3. Sehingga panjang dari barisan observasi T diperoleh dari persamaan (4.1).

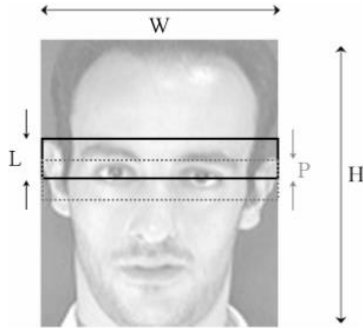
$$T = \frac{H-L}{L-P} + 1 \quad (4.1)$$

Keterangan:

P = Panjang piksel yang *overlap* dari 2 blok

L = Panjang piksel dalam blok

H = Panjang piksel dalam suatu gambar



Gambar 4.3. Barisan *Overlapping Blocks*

Pemilihan ukuran L dan P dapat mempengaruhi keakuratan sistem pengenalan wajah. Presentase ukuran P yang tinggi secara signifikan akan meningkatkan kinerja sistem sehingga meningkatkan kompleksitas komputasi. Selama $P \leq L - 1$ dan $L \approx H/10$, variasi panjang ukuran L tidak mempengaruhi tingkat pengenalan sistem. Ambil contoh untuk matrik ukuran 8×8 $L = 4$ dan $P = 3$ sehingga didapatkan ekstraksi blok seperti berikut:

$$\begin{aligned}
\text{blok_1} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 80 & 63 & 57 & 56 & 54 & 54 & 0 \\ 0 & 80 & 62 & 55 & 55 & 54 & 54 & 0 \\ 0 & 58 & 56 & 54 & 54 & 54 & 53 & 0 \end{bmatrix} \\
\text{blok_2} &= \begin{bmatrix} 0 & 80 & 63 & 57 & 56 & 54 & 54 & 0 \\ 0 & 80 & 62 & 55 & 55 & 54 & 54 & 0 \\ 0 & 58 & 56 & 54 & 54 & 54 & 53 & 0 \\ 0 & 57 & 56 & 54 & 54 & 54 & 53 & 0 \end{bmatrix} \\
\text{blok_3} &= \begin{bmatrix} 0 & 80 & 62 & 55 & 55 & 54 & 54 & 0 \\ 0 & 58 & 56 & 54 & 54 & 54 & 53 & 0 \\ 0 & 57 & 56 & 54 & 54 & 54 & 53 & 0 \\ 0 & 55 & 52 & 52 & 52 & 52 & 51 & 0 \end{bmatrix} \\
\text{blok_4} &= \begin{bmatrix} 0 & 58 & 56 & 54 & 54 & 54 & 53 & 0 \\ 0 & 57 & 56 & 54 & 54 & 54 & 53 & 0 \\ 0 & 55 & 52 & 52 & 52 & 52 & 51 & 0 \\ 0 & 55 & 52 & 52 & 52 & 51 & 51 & 0 \end{bmatrix} \\
\text{blok_5} &= \begin{bmatrix} 0 & 57 & 56 & 54 & 54 & 54 & 53 & 0 \\ 0 & 55 & 52 & 52 & 52 & 52 & 51 & 0 \\ 0 & 55 & 52 & 52 & 52 & 51 & 51 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}
\end{aligned}$$

4.4 Ekstraksi Fitur

Setelah ditetapkan *block overlapping*-nya maka masing-masing blok akan dikenakan algoritma SVD sehingga setiap blok yang awalnya berukuran $W \times L$ akan direpresentasikan dengan 3 nilai pada masing-masing blok. Seperti yang telah dijabarkan sebelumnya dalam persamaan (2.3), SVD mengandung 3 matrik (U, S dan V). Proses perhitungan SVD untuk blok_1 adalah sebagai berikut:

$$X = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 80 & 63 & 57 & 56 & 54 & 54 & 0 \\ 0 & 80 & 62 & 55 & 55 & 54 & 54 & 0 \\ 0 & 58 & 56 & 54 & 54 & 54 & 53 & 0 \end{bmatrix}$$

$$X^T = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 80 & 80 & 58 \\ 0 & 63 & 62 & 56 \\ 0 & 57 & 55 & 54 \\ 0 & 56 & 55 & 54 \\ 0 & 54 & 54 & 54 \\ 0 & 54 & 54 & 53 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$X X^T = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 22586 & 22353 & 20048 \\ 0 & 22353 & 22126 & 19830 \\ 0 & 20048 & 19830 & 18057 \end{bmatrix}$$

Selanjutnya mencari *Eigenvalue* dari $X X^T$

$$\det |X X^T - \lambda I| = 0$$

$$\det \begin{vmatrix} 0 - \lambda & 0 & 0 & 0 \\ 0 & 22586 - \lambda & 22353 & 20048 \\ 0 & 22353 & 22126 - \lambda & 19830 \\ 0 & 20048 & 19830 & 18057 - \lambda \end{vmatrix} = 0$$

$$\lambda_1 = 0, \lambda_2 = 2, \lambda_3 = 195, \lambda_4 = 62573$$

Eigenvektor dari $X X^T$ adalah

$$U = \begin{vmatrix} 0 & 0 & 0 & 1.0000 \\ -0.6005 & -0.3521 & -0.7180 & 0 \\ -0.5942 & -0.4044 & 0.6953 & 0 \\ -0.5351 & 0.8441 & 0.0336 & 0 \end{vmatrix}$$

Selanjutnya dicari *eigenvalue* dan *eigenvektor* dari

$$X^T X = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 16164 & 13248 & 12092 & 12012 & 11772 & 11714 & 0 \\ 0 & 13248 & 10949 & 10025 & 9962 & 9774 & 9718 & 0 \\ 0 & 12092 & 10025 & 9190 & 9133 & 8964 & 8910 & 0 \\ 0 & 12012 & 9962 & 9133 & 9077 & 8910 & 8856 & 0 \\ 0 & 11772 & 9774 & 8964 & 8910 & 8748 & 8694 & 0 \\ 0 & 11714 & 9718 & 8910 & 8856 & 8694 & 8641 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Dengan cara yang sama untuk mencari *eigenvalue* dan *eigenvektor* dari $X X^T$ maka didapatkan *eigenvalue* $\lambda_1 = 0$, $\lambda_2 = 2$, $\lambda_3 = 195$, $\lambda_4 = 62573$ dan *eigenvektor* nya adalah

$$V = \begin{vmatrix} 0 & 0 & 0 & 1.0000 & 0 & 0 & 0 & 0 \\ -0.5062 & -0.8278 & 0.1083 & 0 & 0.0107 & -0.1352 & -0.1685 & 0 \\ -0.4183 & 0.0012 & -0.1983 & 0 & -0.0871 & 0.6513 & 0.5950 & 0 \\ -0.3830 & 0.2343 & -0.7077 & 0 & -0.3916 & -0.2818 & -0.2545 & 0 \\ -0.3806 & 0.2595 & -0.1238 & 0 & 0.8703 & -0.0877 & -0.0860 & 0 \\ -0.3734 & 0.3389 & 0.4786 & 0 & -0.2125 & 0.3919 & -0.5638 & 0 \\ -0.3713 & 0.2785 & 0.4513 & 0 & -0.1908 & -0.5630 & 0.4771 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0000 \end{vmatrix}$$

Sedangkan matrik S adalah

$$S = \begin{bmatrix} \sqrt{62573} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sqrt{195} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sqrt{2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 250.1450 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 13.9631 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1.2295 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} 250.1450 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 13.9631 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1.2295 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Sehingga didapatkan SVD dari matrik blok_1

adalah $U = \begin{vmatrix} 0 & 0 & 0 & 1.0000 \\ -0.6005 & -0.3521 & -0.7180 & 0 \\ -0.5942 & -0.4044 & 0.6953 & 0 \\ -0.5351 & 0.8441 & 0.0336 & 0 \end{vmatrix}$

$$S = \begin{bmatrix} 250.1450 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 13.9631 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1.2295 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$V = \begin{vmatrix} 0 & 0 & 0 & 1.0000 & 0 & 0 & 0 & 0 \\ -0.5062 & -0.8278 & 0.1083 & 0 & 0.0107 & -0.1352 & -0.1685 & 0 \\ -0.4183 & 0.0012 & -0.1983 & 0 & -0.0871 & 0.6513 & 0.5950 & 0 \\ -0.3830 & 0.2343 & -0.7077 & 0 & -0.3916 & -0.2818 & -0.2545 & 0 \\ -0.3806 & 0.2595 & -0.1238 & 0 & 0.8703 & -0.0877 & -0.0860 & 0 \\ -0.3734 & 0.3389 & 0.4786 & 0 & -0.2125 & 0.3919 & -0.5638 & 0 \\ -0.3713 & 0.2785 & 0.4513 & 0 & -0.1908 & -0.5630 & 0.4771 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0000 \end{vmatrix}$$

Dalam penelitian ini koefisien yang diambil adalah 2 koefisien matrik S dan koefisien pertama matrik U sebagai 3 fitur (U_{11}, S_{11}, S_{22}) . [12] Untuk blok_1 sampai blok_5 nilai fiturnya adalah

$$\begin{aligned} [U_{11} \ S_{11} \ S_{22}]_1 &= [\ 0 \quad 250.1450 \ 13.9631] \\ [U_{11} \ S_{11} \ S_{22}]_2 &= [-0.5293 \ 283.5326 \ 17.8536] \\ [U_{11} \ S_{11} \ S_{22}]_3 &= [-0.5434 \ 272.5835 \ 16.2272] \\ [U_{11} \ S_{11} \ S_{22}]_4 &= [-0.5124 \ 262.2524 \ 1.6851] \\ [U_{11} \ S_{11} \ S_{22}]_5 &= [-0.5947 \ 225.2120 \ 1.4636] \end{aligned}$$

Kemudian koefisien SVD akan membangun vektor observasi $X = (x_1, x_2, \dots, x_n)$. Tetapi koefisien tersebut belum bisa dimodelkan oleh HMM sehingga perlu dilakukan kuantisasi vektor. Hal ini biasa dilakukan dengan proses pembulatan, pemotongan, atau proses yang lain. Misalkan sebuah barisan akan dikuantisasi ke dalam tingkat yang berbeda, sehingga perbedaan antara dua nilai kuantitatif akan menjadi persamaan (4.2).

$$\Delta_i = (x_{i\max} - x_{i\min}) / D_i \quad (4.2)$$

$$\Delta_i = \frac{([0 \ 283.5326 \ 17.8536] - [-0.5947 \ 225.2120 \ 1.4636])}{[18 \ 10 \ 7]}$$

$$= [0.33 \ 6.0321 \ 2.3413]$$

Dimana perbedaan antara dua nilai (D_i) yang digunakan untuk kuantisasi koefisien U(1,1), S(1,1), S(2,2) adalah 18, 10 dan 7. [12] Sehingga nilai label mempunyai 1260 kemungkinan vektor yang berbeda untuk tiap blok.

Setelah diketahui Δ_i komponen x_i akan digantikan dengan nilai kuantisasi sebagai berikut:

$$q_t = \frac{x_i - x_i^{min}}{\Delta_i} \quad (4.3)$$

Untuk blok_1 sampai blok_5 nilai kuantisasinya adalah sebagai berikut :

$$\begin{aligned} q_1 &= \frac{[0 \quad 250.1450 \quad 13.9631] - [-0.5947 \quad 225.2120 \quad 1.4636]}{[0.33 \ 6.0321 \ 2.3413]} \\ &= [1 \quad 4 \quad 5] \\ q_2 &= [0 \quad 9 \quad 7] \\ q_3 &= [0 \quad 7 \quad 6] \\ q_4 &= [0 \quad 6 \quad 0] \\ q_5 &= [1 \quad 1 \quad 1] \end{aligned}$$

Tahap selanjutnya yakni pelabelan. Proses ini bertujuan untuk merepresentasikan setiap blok dengan satu nilai diskrit dengan rumus:

$$Label = q_{t_1} * 10 * 7 + q_{t_2} * 7 + q_{t_3} + 1 \quad (4.4)$$

Untuk blok_1 sampai blok_5 nilai pelabelannya adalah sebagai berikut :

$$\begin{aligned} Label_1 &= 1 * 10 * 7 + 4 * 7 + 5 + 1 = 104 \\ Label_2 &= 0 * 10 * 7 + 9 * 7 + 7 + 1 = 71 \\ Label_3 &= 0 * 10 * 7 + 7 * 7 + 6 + 1 = 56 \\ Label_4 &= 0 * 10 * 7 + 6 * 7 + 0 + 1 = 43 \\ Label_5 &= 1 * 10 * 7 + 1 * 7 + 1 + 1 = 79 \end{aligned}$$

Sehingga untuk barisan vektor observasinya adalah $O = (104,71,56,43,79)$.

4.5 Proses Training

Proses *training* memerlukan input berupa barisan vektor observasi. Proses *training* diawali dengan membuat *prototype* HMM model $\lambda = (A, B, \pi)$. Proses pengenalan menggunakan wajah yang dibagi menjadi 5 bagian yaitu: Dahi, mata, hidung, mulut, dan dagu. Dengan asumsi sebuah blok yang bergerak dari atas ke bawah dan blok tersebut menunjukkan salah satu dari 5 state secara berurutan. Sehingga, misalnya blok berada di state “mata” maka blok tersebut tidak akan pernah menjadi “mulut”, ia akan berjalan menuju “hidung”. Oleh karena itu probabilitas bergerak dari satu blok ke blok berikutnya adalah 50 % dan tetap berada di blok yang sama juga 50%.

Keadaan awal suatu sistem selalu berada pada kepala dengan probabilitas 1. Dengan demikian parameter awal adalah $\pi = [1 \ 0 \ 0 \ 0 \ 0]$.

Untuk matrik probabilitas transisi state adalah

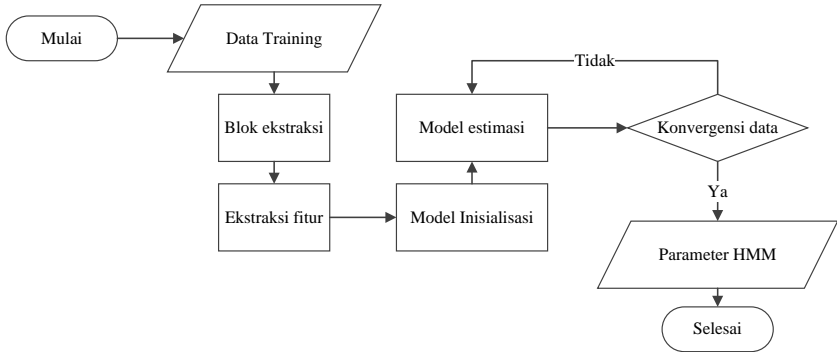
$$A = \begin{bmatrix} 0.5 & 0.5 & 0 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 & 0 \\ 0 & 0 & 0.5 & 0.5 & 0 \\ 0 & 0 & 0 & 0.5 & 0.5 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Dan untuk matrik probabilitas observasi adalah

$$B = \frac{1}{1260} \begin{bmatrix} 1 & \dots & 1260 \\ 1 & \begin{bmatrix} 1 & \dots & 1 \end{bmatrix} \\ \vdots & \begin{bmatrix} \vdots & \ddots & \vdots \end{bmatrix} \\ 5 & \begin{bmatrix} 1 & \dots & 1 \end{bmatrix} \end{bmatrix}$$

Kemudian setiap model HMM awal di *training* ke setiap individu yang ada di database menggunakan algoritma Baum-Welch[16] sehingga akan mendapatkan $\lambda =$

$\max_{\lambda} |\lambda^{(k)}|$. Proses Training dapat dilihat pada Gambar 4.4. Dalam algoritma Baum-Welch, didefinisikan empat variabel, yaitu: variabel forward, variabel backward, variabel $\xi_t(i,j)$ dan variabel $\gamma_t(i)$. Variabel forward dan variabel backward akan digunakan dalam perhitungan variabel $\xi_t(i,j)$ dan variabel $\gamma_t(i)$.



Gambar 4.4. Diagram Proses Training

Variabel forward didapatkan dari algoritma Forward dengan langkah sebagai berikut:

1. Inisialisasi

$$\alpha_1(i) = \pi_i b_i(O_1) \quad (4.5)$$

$$\begin{aligned} \alpha_1(1) &= \pi_1 b_1(O_1) = (1)(0.00079365) \\ &= 0.00079365 \end{aligned}$$

$$\alpha_1(2) = \pi_2 b_2(O_2) = (0)(0.00079365) = 0$$

$$\alpha_1(3) = \pi_3 b_3(O_3) = (0)(0.00079365) = 0$$

$$\alpha_1(4) = \pi_4 b_4(O_4) = (0)(0.00079365) = 0$$

$$\alpha_1(5) = \pi_5 b_5(O_5) = (0)(0.00079365) = 0$$

2. Tahap induksi

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^N \alpha_t(i) a_{ij} \right] b_j(O_{t+1}) \quad (4.6)$$

$$t = 1$$

$$\begin{aligned}
\alpha_2(1) &= \left[\sum_{i=1}^N \alpha_1(i) a_{i1} \right] b_1(O_2) \\
&= [(0.00079365)(0.5) + (0)(0) \\
&\quad + (0)(0) + (0)(0) \\
&\quad + (0)(0)](0.00079365) \\
&= 3,1494e - 07
\end{aligned}$$

$$\begin{aligned}
\alpha_2(2) &= [(0.00079365)(0.5) + (0)(0.5) + (0)(0) \\
&\quad + (0)(0) + (0)(0)](0.00079365) \\
&= 3,1494e - 07
\end{aligned}$$

$$\begin{aligned}
\alpha_2(3) &= [(0.00079365)(0) + (0)(0.5) + (0)(0.5) \\
&\quad + (0)(0) + (0)(0)](0.00079365) = 0
\end{aligned}$$

$$\begin{aligned}
\alpha_2(4) &= [(0.00079365)(0) + (0)(0) + (0)(0.5) \\
&\quad + (0)(0.5) + (0)(0)](0.00079365) \\
&= 0
\end{aligned}$$

$$\begin{aligned}
\alpha_2(5) &= [(0.00079365)(0) + (0)(0) + (0)(0) \\
&\quad + (0)(0.5) + (0)(1)](0.00079365) \\
&= 0
\end{aligned}$$

$$t = 2$$

$$\alpha_3(1) = 1,2498e - 10$$

$$\alpha_3(2) = 2,4995e - 10$$

$$\alpha_3(3) = 1,2498e - 10$$

$$\alpha_3(4) = 0$$

$$\alpha_3(5) = 0$$

$$t = 3$$

$$\alpha_4(1) = 4,9594e - 14$$

$$\alpha_4(2) = 1,4878e - 13$$

$$\alpha_4(3) = 1,4878e - 13$$

$$\alpha_4(4) = 4,9594e - 14$$

$$\alpha_4(5) = 0$$

$$t = 4$$

$$\alpha_5(1) = 1,9680e - 17$$

$$\alpha_5(2) = 7,8720e - 17$$

$$\alpha_5(3) = 1,1808e - 16$$

$$\alpha_5(4) = 7,8720e - 17$$

$$\alpha_5(5) = 1,9680e - 17$$

Sehingga didapatkan hasil yaitu

$$\alpha = \begin{bmatrix} 7,9370e - 04 & 0 & 0 & 0 & 0 \\ 3,1494e - 07 & 3,1494e - 07 & 0 & 0 & 0 \\ 1,2498e - 10 & 2,4995e - 10 & 1,2498e - 10 & 0 & 0 \\ 4,9594e - 14 & 1,4878e - 13 & 1,4878e - 13 & 4,9594e - 14 & 0 \\ 1,9680e - 17 & 7,8720e - 17 & 1,1808e - 16 & 7,8720e - 17 & 1,9680e - 17 \end{bmatrix}$$

3. Tahap terminasi

$$P(O|\lambda) = \sum_{i=1}^N \alpha_T(i) \quad (4.7)$$

$$\begin{aligned} P(O|\lambda) &= \sum_{i=1}^N \alpha_5(i) \\ &= 1,9680e - 17 + 7,8720e - 17 \\ &\quad + 1,1808e - 16 + 7,8720e - 17 \\ &\quad + 1,9680e - 17 = 3.1488e - 16 \end{aligned}$$

Variabel *Backward* didapatkan dari algoritma *Backward* dengan langkah sebagai berikut:

1. Tahap inisialisasi

$$\begin{aligned} \beta_T(i) &= 1 \text{ untuk } 1 \leq i \leq N \\ \beta_5(1) &= \beta_5(2) = \beta_5(3) = \beta_5(4) = 1 \end{aligned} \quad (4.8)$$

2. Tahap induksi

$$\beta_t(i) = \sum_{j=1}^N b_j(O_{t+1})\beta_{t+1}(j) a_{ij} \quad \text{untuk } T-1 \leq t \leq 1 \text{ dan } 1 \leq i \leq N. \quad (4.9)$$

$$t = 4$$

$$\begin{aligned} \beta_4(1) &= \sum_{j=1}^N b_j(O_5)\beta_5(j) a_{1j} \\ &= (0.00079365)(1)(0.5) \\ &\quad + (0.00079365)(1)(0.5) \\ &\quad + (0.00079365)(1)(0) \\ &\quad + (0.00079365)(1)(0) \\ &\quad + (0.00079365)(1)(0) \\ &= 7.9365e - 4 \\ \beta_4(2) &= (0.00079365)(1)(0) \\ &\quad + (0.00079365)(1)(0.5) \\ &\quad + (0.00079365)(1)(0.5) \\ &\quad + (0.00079365)(1)(0) \\ &\quad + (0.00079365)(1)(0) \\ &= 7.9365e - 4 \\ \beta_4(3) &= (0.00079365)(1)(0) \\ &\quad + (0.00079365)(1)(0) \\ &\quad + (0.00079365)(1)(0.5) \\ &\quad + (0.00079365)(1)(0.5) \\ &\quad + (0.00079365)(1)(0) \\ &= 7.9365e - 4 \\ \beta_4(4) &= (0.00079365)(1)(0) \\ &\quad + (0.00079365)(1)(0) \\ &\quad + (0.00079365)(1)(0) \\ &\quad + (0.00079365)(1)(0.5) \\ &\quad + (0.00079365)(1)(0.5) \\ &= 7.9365e - 4 \end{aligned}$$

$$\begin{aligned}
\beta_4(5) &= (0.00079365)(1)(0) \\
&\quad + (0.00079365)(1)(0) \\
&\quad + (0.00079365)(1)(0) \\
&\quad + (0.00079365)(1)(0) \\
&\quad + (0.00079365)(1)(1) \\
&= 7.9365e - 4
\end{aligned}$$

$$t = 3$$

$$\beta_3(1) = 6,2988e - 07$$

$$\beta_3(2) = 6,2988e - 07$$

$$\beta_3(3) = 6,2988e - 07$$

$$\beta_3(4) = 6,2988e - 07$$

$$\beta_3(5) = 6,2988e - 07$$

$$t = 2$$

$$\beta_2(1) = 4,9991e - 10$$

$$\beta_2(2) = 4,9991e - 10$$

$$\beta_2(3) = 4,9991e - 10$$

$$\beta_2(4) = 4,9991e - 10$$

$$\beta_2(5) = 4,9991e - 10$$

$$t = 1$$

$$\beta_1(1) = 3,9675e - 13$$

$$\beta_1(2) = 3,9675e - 13$$

$$\beta_1(3) = 3,9675e - 13$$

$$\beta_1(4) = 3,9675e - 13$$

$$\beta_1(5) = 3,9675e - 13$$

Sehingga didapatkan hasil yaitu

$$\beta = \begin{bmatrix} 3,9675e - 13 & 3,9675e - 13 & 3,9675e - 13 & 3,9675e - 13 & 3,9675e - 13 \\ 4,9991e - 10 & 4,9991e - 10 & 4,9991e - 10 & 4,9991e - 10 & 4,9991e - 10 \\ 6,2988e - 07 & 6,2988e - 07 & 6,2988e - 07 & 6,2988e - 07 & 6,2988e - 07 \\ 7,9365e - 4 & 7,9365e - 4 & 7,9365e - 4 & 7,9365e - 4 & 7,9365e - 4 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

3. Tahap terminasi

$$P(O|\lambda) = \sum_{i=1}^N b_i(O_1) \pi(i) \beta_1(i) \quad (4.10)$$

$$\begin{aligned} P(O|\lambda) &= (7.9365e - 4)(3,9675e - 13)(0.5) \\ &\quad + (7.9365e - 4)(3,9675e - 13)(0.5) \\ &\quad + (7.9365e - 4)(3,9675e - 13)(0) \\ &\quad + (7.9365e - 4)(3,9675e - 13)(0) \\ &\quad + (7.9365e - 4)(3,9675e - 13)(0) \\ &= 3.1488e - 16 \end{aligned}$$

Selanjutnya untuk mencari variabel $\xi_t(i,j)$ menggunakan persamaan (4.11):

$$\xi_t(i,j) = \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{P(O|\lambda)} \quad (4.11)$$

$$\begin{aligned} t &= 1 \\ \xi_1(1,1) &= \frac{\alpha_1(1) a_{11} b_1(O_2) \beta_2(1)}{P(O|\lambda)} \\ &= \frac{(0.0007936)(0.5)(0.0007936)(4,9991e - 10)}{3.1488e - 16} \\ &= 0.5000 \end{aligned}$$

$$\begin{aligned} \xi_1(1,2) &= \frac{(0.0007936)(0.5)(0.0007936)(4,9991e - 10)}{3.1488e - 16} \\ &= 0.5000 \end{aligned}$$

$$\xi_1(1,3) = \frac{(0.0007936)(0)(0.0007936)(4,9991e - 10)}{3.1488e - 16}$$

$$= 0$$

$$\xi_1(1,4) = \frac{(0.0007936)(0)(0.0007936)(4,9991e - 10)}{3.1488e - 16}$$

$$= 0$$

$$\xi_1(1,5) = \frac{(0.0007936)(0)(0.0007936)(4,9991e - 10)}{3.1488e - 16}$$

$$= 0$$

$$\begin{array}{llll} \xi_1(2,1) = 0 & \xi_1(3,1) = 0 & \xi_1(4,1) = 0 & \xi_1(5,1) = 0 \\ \xi_1(2,2) = 0 & \xi_1(3,2) = 0 & \xi_1(4,2) = 0 & \xi_1(5,2) = 0 \\ \xi_1(2,3) = 0 & \xi_1(3,3) = 0 & \xi_1(4,3) = 0 & \xi_1(5,3) = 0 \\ \xi_1(2,4) = 0 & \xi_1(3,4) = 0 & \xi_1(4,4) = 0 & \xi_1(5,4) = 0 \\ \xi_1(2,5) = 0 & \xi_1(3,5) = 0 & \xi_1(4,5) = 0 & \xi_1(5,5) = 0 \end{array}$$

Dengan cara yang sama maka akan didapatkan nilai dari variabel $\xi_t(i,j)$ untuk $1 \leq t \leq T-1$, $1 \leq i \leq N$, $1 \leq j \leq N$. Sehingga didapatkan hasil yaitu:

$$\gamma_t(i) = \sum_{j=1}^N \xi_t(i,j)$$

$$\gamma_t(i) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0.5 & 0.5 & 0 & 0 & 0 \\ 0.25 & 0.5 & 0.25 & 0 & 0 \\ 0.125 & 0.375 & 0.375 & 0.125 & 0 \end{bmatrix}$$

Selanjutnya akan dicari estimasi dari parameter awal menggunakan persamaan (4.12), parameter probabilitas transisi menggunakan persamaan (4.13) dan parameter probabilitas observasi menggunakan persamaan (4.14)

$$\hat{\pi} = \gamma_1(i), 1 \leq i \leq N \quad (4.12)$$

Sehingga didapatkan taksiran nilai parameter awal:

$$\hat{\pi} = [1 \quad 0 \quad 0 \quad 0 \quad 0]$$

$$\hat{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)}, 1 \leq i \leq N, 1 \leq j \leq N \quad (4.13)$$

Sehingga didapatkan taksiran nilai probabilitas transisi:

$$\hat{A} = \begin{bmatrix} 0.5 & 0.5 & 0 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 & 0 \\ 0 & 0 & 0.5 & 0.5 & 0 \\ 0 & 0 & 0 & 0.5 & 0.5 \\ NaN & NaN & NaN & NaN & NaN \end{bmatrix}$$

$$\hat{b}_j(k) = \frac{\sum_{t=1}^T O_{t=k} \gamma_t(j)}{\sum_{t=1}^T \gamma_t(j)}, 1 \leq j \leq N, 1 \leq k \leq M \quad (4.14)$$

Sehingga didapatkan taksiran nilai probabilitas observasi dengan ukuran matrik 1x 1260:

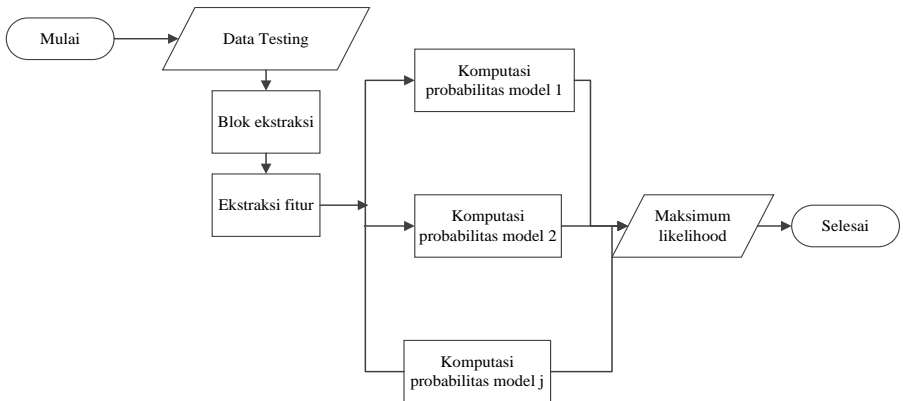
$$\hat{B} = \begin{matrix} & \begin{matrix} 1 & \dots & 43 & \dots & 1260 \end{matrix} \\ \begin{matrix} 1 \\ \vdots \\ 5 \end{matrix} & \begin{bmatrix} 0 & \dots & 0,067 & \dots & 0 \\ \vdots & & \ddots & & \vdots \\ NaN & \dots & & & NaN \end{bmatrix} \end{matrix}$$

Kemudian parameter-parameter HMM tersebut disimpan dalam database yang selanjutnya akan digunakan dalam proses *testing*.

4.6 Proses *Testing*

Setelah dilakukan *training*, maka selanjutnya akan dilakukan *testing* dengan parameter yang dihasilkan dari proses *training* dan juga barisan observasi. Proses *testing* memerlukan input berupa citra dan akan memberikan output berupa hasil pengenalan individu dalam citra. Sebelum dilakukan pengenalan oleh HMM, citra input harus melalui tahap praproses sehingga dihasilkan citra *grayscale* berukuran 64x64 piksel.

Pengenalan dengan HMM yaitu dengan menghitung probabilitas data *testing* menggunakan persamaan (4.5) sampai persamaan (4.7) dengan parameter yang di dapatkan dari data *training* yang telah disimpan sebelumnya. Data *testing* di konversi ke dalam bentuk barisan observasi dan sebuah model *likelihoods* $P(O_{test}|\lambda^{(k)})$. $1 \leq k \leq j$, j = banyak data *training*. Model dengan nilai *likelihood* tertinggi adalah hasil dari data *testing*. Proses *testing* dapat dilihat pada Gambar 4.5.



Gambar 4.5. Diagram Proses *Testing*

BAB V

IMPLEMENTASI SISTEM

Pada bab ini, dibahas mengenai langkah-langkah dalam pengimplemetasian sistem berdasarkan desain sistem yang telah dirancang.

5.1 Lingkungan *Hardware* dan *Software*

Lingkungan perancangan sistem dibangun dari dua lingkungan yaitu lingkungan *software* dan lingkungan *hardware*.

Lingkungan	Spesifikasi	
<i>Hardware</i>	<i>Processor</i>	Intel® Core™ i7-4700HQ CPU @ 2.40GHz
	RAM	4 GB
<i>Software</i>	Sistem Operasi	Windows 10 64-bit
	Tools	MATLAB

5.2 Implementasi *User Interface*

Pada desain *user interface* ini terdapat Halaman Awal, Halaman *Training*, Halaman *Testing* dan Halaman *Testing Video*.

1. Halaman Utama

Halaman utama merupakan antarmuka yang berisi menu-menu untuk menampilkan antarmuka-antarmuka lainnya dalam sistem. Hasil implementasi halaman utama dapat dilihat pada Gambar 5.1.



Gambar 5.1. Halaman Awal

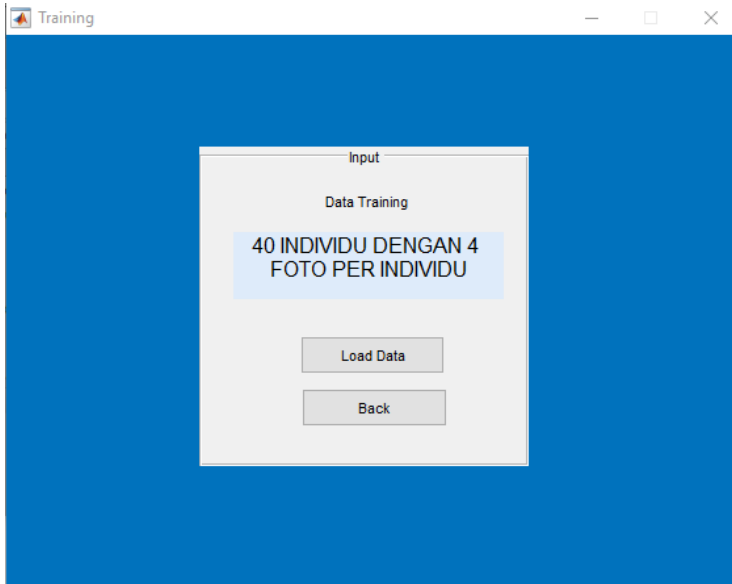
Pada desain Halaman Awal ini terdapat *menu bar* yang berisi sederetan menu-menu yang digunakan oleh sistem. Adapun kegunaan menu-menu yang ditampilkan pada *menu bar* disajikan dalam Tabel 2.

Tabel 1. Kegunaan Menu-Menu dalam Menu Bar

Menu	Kegunaan
<i>Training</i>	Melakukan pelatihan agar sistem dapat menyimpan parameter <i>Training</i> .
<i>Testing</i>	Melakukan pengujian terhadap citra yang sudah di <i>Training</i> dan citra <i>Testing</i> . Citra berupa gambar individu.
<i>Testing Video</i>	Melakukan ekstraksi frame terhadap video. Kemudian melakukan <i>Testing</i> terhadap frame tersebut.

2. Halaman *Training*

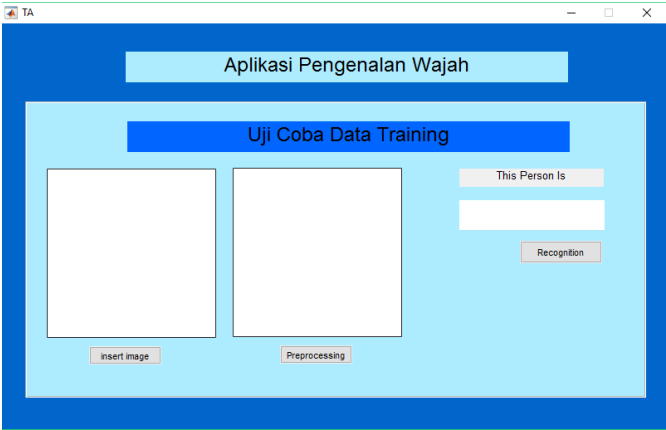
Halaman *Training* merupakan antarmuka yang berisi tombol *Load Data* untuk melakukan *Training* data dan tombol *Back* untuk kembali ke Halaman Awal. Hasil implementasi halaman utama dapat dilihat pada Gambar 5.2.



Gambar 5.2. Halaman *Training*

3. Halaman *Testing*

Halaman *Testing* merupakan antarmuka yang berguna untuk melakukan uji pengenalan terhadap citra. Hasil implementasi halaman utama dapat dilihat pada Gambar 5.3.



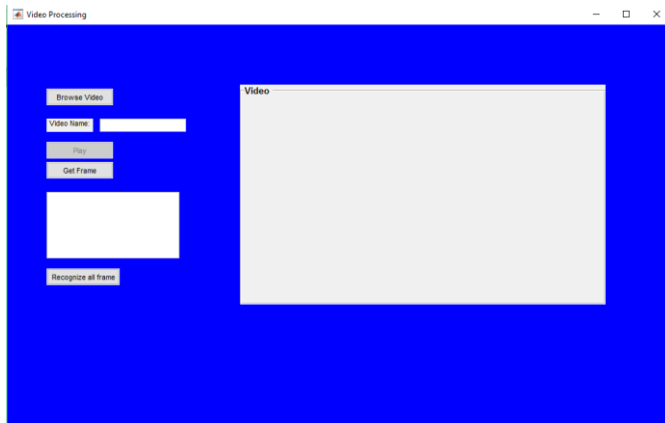
Gambar 5.3. Halaman *Testing*

Tabel 2. Kegunaan Tombol dalam Halaman *Testing*

Menu	Kegunaan
<i>Insert Image</i>	Memasukkan citra yang akan di uji. Citra berupa gambar yang diambil dari direktori file komputer.
<i>Preprocessing</i>	Melakukan <i>preprocessing</i> terhadap citra yang telah dimasukkan.
<i>Recognition</i>	Melakukan pengenalan terhadap cutra yang telah dimasukkan. Nama individu akan muncul dalam kotak diatas tombol <i>Recognition</i>

4. Halaman *Testing Video*

Halaman *Testing Video* merupakan antarmuka yang berguna untuk melakukan uji pengenalan terhadap video. Hasil implementasi halaman utama dapat dilihat pada Gambar 5.4.



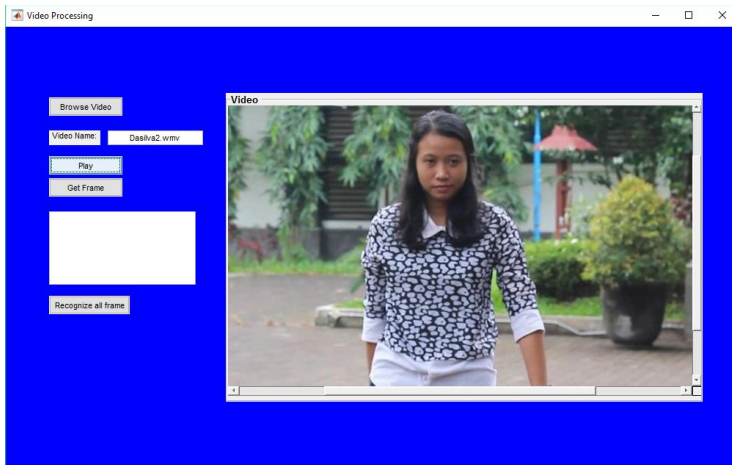
Gambar 5.4. Halaman *Testing Video*

Menu	Kegunaan
<i>Browse Video</i>	Memasukkan video yang akan di proses.
<i>Play</i>	Menampilkan video dalam GUI.
<i>Get Frame</i>	Melakukan ekstraksi frame terhadap video. Kemudian frame tersebut tersimpan dalam direktori komputer.
<i>Recognition Frame</i>	Melakukan pengenalan terhadap frame yang sudah diekstrak dari video

Hasil running program dapat dilihat pada Gambar 5.5 sampai Gambar 5.6.



Gambar 5.5. Hasil *Running Program*



Gambar 5.6. Hasil *Running Program*

5.3 Implementasi Praproses Data

Praproses ini terdiri dari proses deteksi wajah, *cropping*, konversi citra warna ke *grayscale*, penyesuaian

ukuran citra dan *filtering*. Algoritma deteksi wajah yang digunakan adalah algoritma standar Paul Viola dan Michael Jones. Setelah dilakukan deteksi wajah kemudian wajah yang terdeteksi akan diberikan *bounding box* dan kemudian dilakukan proses *cropping* sesuai ukuran *bounding box*. Hasil dari deteksi wajah dapat dilihat pada Gambar 5.7 dan Gambar 5.8. Implementasi program untuk deteksi tepi adalah sebagai berikut :

```
I = imread(['./Data wajah training/']);  
faceDetector = vision.CascadeObjectDetector;  
bboxes = step(faceDetector,I);  
handles.IFaces = insertObjectAnnotation(I,  
'rectangle', bboxes, 'Face');  
I3 = imcrop(I,bboxes);
```



Gambar 5.7. Gambar wajah yang akan di proses



Gambar 5.8. Gambar setelah Proses *Cropping*

Selanjutnya citra berwarna yang telah melalui proses *cropping* akan dilakukan penyamaan ukuran citra dan dikonversi menjadi citra *grayscale*. Kemudian citra *grayscale* tersebut dilakukan filter dengan menggunakan *minimum order-statistic filtering* seperti pada Gambar 5.9.

Implementasi program untuk proses penyeragaman ukuran, konversi dan filtering adalah sebagai berikut :

```
a=imresize(I3,[64,64]);
Im = rgb2gray(a);
or = ordfilt2(Im,1,true(3));
```



a

b

Gambar 5.9. Setelah Proses *Grayscale* (a) dan setelah Proses *filtering* (b)

5.4 Implementasi Ekstraksi Fitur

Selanjutnya citra hasil *filtering* di bagi menjadi beberapa blok sesuai ukuran T. T dihitung melalui persamaan 4.1. Untuk masing – masing blok dilakukan ekstraksi fitur menggunakan algoritma SVD. Berikut adalah implementasi untuk ekstraksi blok dan juga ekstraksi fitur SVD:

```
blk_index = 0;
for blk_begin=1:52
    blk_index=blk_index+1;
    g = ordfilt2(Im,1,true(3));
    blk = g(blk_begin:blk_begin+4,:);
    [U,S,V] = svd(double(blk));
    blk_coeffs = [U(1,1) S(1,1) S(2,2)];
    max_coeffs = max([max_coeffs;blk_coeffs]);
    min_coeffs = min([min_coeffs;blk_coeffs]);
    blk_cell{blk_index,face_index} =
    blk_coeffs;
end
```

Setelah didapatkan ekstraksi fitur setiap blok maka dilakukan kuantisasi vektor dan pelabelan untuk mendapatkan 1D barisan observasi. Implementasi vektor kuantisasi dan pelabelan adalah sebagai berikut :

```
Min_coeffs = minmax(1,:);
delta_coeffs = minmax(3,:);
qt = floor((blk_coeffs-
min_coeffs)./delta_coeffs);
label = qt(1)*18+qt(2)*10+qt(3)*7;
```

5.5 Implementasi proses *Training*

Proses *training* memerlukan input berupa barisan vektor observasi yang didapatkan dari proses ekstraksi fitur dan juga inisialisasi parameter HMM yang didapatkan dari hasil implementasi sebagai berikut:

```
transmat1 = ones(5,5) * eps;
transmat1(5,5) = 1;
for r=1:6
    transmat1(r,r) = 0.5;
```

```

transmat1(r,r+1) = 0.5;
end
obsmat1 = (1/1260)*ones(5,1260);

```

Setelah didapatkan inisialisasi parameter Mencari parameter – parameter awal dalam HMM maka dilanjutkan dengan mencari parameter-parameter yang optimal untuk mendapatkan peluang observasi yang maksimum dengan algoritma Baum-Welch. Implementasinya adalah sebagai berikut :

```

[ESTTR, ESTEMIT]=hmmtrain(seqmat,transmat1,obs
mat1, 'Algorithm', 'BaumWelch');
mESTTR = max(ESTTR,eps);
mESTEMIT = max(ESTEMIT,eps);

```

5.6 Implementasi Proses *Testing*

Pengenalan dengan HMM yaitu dengan mencocokkan data *testing* dengan hasil data *training* yang telah disimpan sebelumnya. Pada tahap ini menggunakan algoritma forward untuk mendapatkan likelihood :

```

for l=1:number_of_persons_in_database
    TRANS = Database{6,1}{1,1};
    EMIS = Database{6,1}{1,2};
    PRIOR = [0.2 0.2 0.2 0.2 0.2];
n=length(TRANS(1,:));
T=length(seq);
for i=1:n           %Inisialisasi
    m(1,i)=EMIS(i,seq(1))*PRIOR(i);
end
for t=1:(T-1)       %Rekursi
    for j=1:n
        z=0;
        for i=1:n
            z=z+TRANS(i,j)*m(t,i);
        end
        m(t+1,j)=z*EMIS(j,seq(t+1));
    end
end
p=0;

```

```

for i=1:n           %Terminasi
    p=p+m(T,i);
end
    results(1,1) = p;
end

```

Setelah didapatkan likelihood pada masing-masing data dari database maka dilakukan pencarian likelihood tertinggi untuk menentukan hasil pengenalan. Implementasinya adalah sebagai berikut

```

[max, person_index] = max(results);
fprintf(['This person is
', myDatabase{1, person_index}, '.\n']);

```

Implementasi program lebih lengkap dapat dilihat dalam LAMPIRAN 2.

BAB VI

UJI COBA DAN EVALUASI SISTEM

Pada bab ini dijelaskan tahap-tahap uji coba berdasarkan implementasi sistem yang telah dibuat beserta pengujian kinerja sistem. Pengujian dilakukan terhadap dataset yang disimpan di dalam direktori komputer dan akan dihitung tingkat akurasi dari sistem.

6.1 Dataset Uji Coba *Training*

Dataset uji coba *training* terdiri dari 160 citra wajah dari 40 individu yang berbeda. Masing-masing individu terdapat 4 citra wajah yang dilakukan proses *training*. Dataset dari data uji coba *training* ditampilkan pada Tabel 3.

Tabel 3. Data *Training*

Dataset Uji Coba <i>Training</i>				
Jumlah	40 x 4 gambar wajah			
Relawan	Adit Afif Agus Alfin Amel Anas Anin Anshar Ardi Ardif	Aufa Ayu Bhara Bryan Corry Dani Dasilva Dian Dita Fadhlan	Faizin Fella Feri Gery Hayu Indra Intan Isabella Ivan Khoirul	Lingga Nadya Nirwana Nuke Romli Rozi Satria Sifaul Uyun Wawan
Tipe	Berwarna			
State	<pre> graph LR Dahi((Dahi)) --> Dahi Mata((Mata)) --> Mata Hidung((Hidung)) --> Hidung Mulut((Mulut)) --> Mulut Dag((Dag)) --> Dag Dahi --> Mata Mata --> Hidung Hidung --> Mulut Mulut --> Dag </pre>			

6.2 Dataset Uji Coba Proses *Testing*

Dataset uji coba *testing* dibedakan menjadi 2 yaitu citra wajah bergerak yang terdiri dari 40 citra wajah yang diambil dari frame video dan citra wajah yang tidak bergerak yang terdiri dari 80 citra wajah.

6.3 Hasil Uji Coba Pengenalan

Pengujian sistem menggunakan citra dari 40 individu yang berbeda yang terdiri dari 160 citra data *training* dan 120 citra data *testing*. Pengujian proses ini untuk menentukan siapa pemilik dari wajah tersebut.

6.3.1. Uji Coba 1

Uji coba 1 menggunakan penyamaan ukuran citra 64x64 piksel dan barisan observasi $T = \frac{64-6}{6-5} + 1 = 59$. Hasil dari uji coba 1 dapat dilihat pada Tabel 4 dan Tabel 5.

Tabel 4. Hasil Uji Coba 1 dengan Data *Training*

Uji Coba Data <i>Training</i>	
Jumlah Data	160
Benar	157
Salah	3
Akurasi	98,13%

Tabel 5. Hasil Uji Coba 1 dengan Data *Testing*

Uji Coba Data <i>Testing</i>			
	Individu tidak bergerak	Individu bergerak	Akurasi Total
Jumlah Data	80	40	
Benar	68	29	80,8%
Salah	12	11	
Akurasi	85%	72,5%	

6.3.2. Uji Coba 2

Uji coba 2 menggunakan penyamaan ukuran citra 64x64 piksel dan barisan observasi $T = \frac{64-7}{7-6} + 1 = 57$. Hasil dari uji coba 2 dapat dilihat pada Tabel 6 dan Tabel 7.

Tabel 6. Hasil Uji Coba 2 dengan Data Training

Uji Coba Data Training	
Jumlah Data	160
Benar	154
Salah	6
Akurasi	96,25%

Tabel 7. Hasil Uji Coba 2 dengan Data Testing

Uji Coba Data Testing			
	Individu tidak bergerak	Individu bergerak	Akurasi Total
Jumlah Data	80	40	
Benar	68	20	73,3%
Salah	12	20	
Akurasi	85%	50%	

6.3.3. Uji Coba 3

Uji coba 3 menggunakan penyamaan ukuran citra 128x128 piksel dan barisan observasi $T = \frac{128-12}{12-11} + 1 = 117$. Hasil dari uji coba 3 dapat dilihat pada Tabel 8 dan Tabel 9.

Tabel 8. Hasil Uji Coba 3 dengan Data Training

Uji Coba Data Training	
	Training
Jumlah Data	160
Benar	149
Salah	11
Akurasi	93,13%

Tabel 9. Hasil Uji Coba 3 dengan Data Testing

Uji Coba Data Testing			
	Individu tidak bergerak	Individu bergerak	Akurasi Total
Jumlah Data	80	40	
Benar	53	20	60,8%
Salah	27	20	
Akurasi	66,25%	50%	

6.3.4. Uji Coba 4

Uji coba 4 menggunakan penyamaan ukuran citra 128x128 piksel dan barisan observasi $T = \frac{128-13}{13-12} + 1 = 116$. Hasil dari uji coba 4 dapat dilihat pada Tabel 10 dan Tabel 11.

Tabel 10. Hasil Uji Coba 4 dengan Data Training

Uji Coba Data Training	
	Training
Jumlah Data	160
Benar	151
Salah	8
Akurasi	94,38%

Tabel 11. Hasil Uji Coba 4 dengan Data *Testing*

Uji Coba Data <i>Testing</i>			
	Individu tidak bergerak	Individu bergerak	Akurasi Total
Jumlah Data	80	40	
Benar	50	18	46%
Salah	30	2	
Akurasi	62,5%	45%	

6.3.5. Uji Coba 5

Uji coba 5 menggunakan penyamaan ukuran citra 256x256 piksel dan barisan observasi $T = \frac{256-25}{25-24} + 1 = 232$. Hasil dari uji coba 5 dapat dilihat pada Tabel 12 dan Tabel 13.

Tabel 12. Hasil Uji Coba 5 dengan Data *Training*

Uji Coba Data <i>Training</i>	
	Training
Jumlah Data	160
Benar	99
Salah	61
Akurasi	61,88%

Tabel 13. Hasil Uji Coba 5 dengan Data *Testing*

Uji Coba Data <i>Testing</i>			
	Individu tidak bergerak	Individu bergerak	Akurasi Total
Jumlah Data	80	40	
Benar	28	0	23,3%
Salah	52	0	
Akurasi	35%	0%	

6.4 Pembahasan Hasil Pengujian

Pembahasan hasil pengujian difokuskan pada hasil pengujian dan proses pengenalan individu melalui identifikasi wajah. Pada penelitian ini pembahasan hasil pengujian digunakan untuk mengetahui kinerja sistem pengenalan individu melalui identifikasi wajah menggunakan Metode *Hidden Markov Model*. Dari hasil yang didapatkan pada subbab sebelumnya, sistem mampu mengenali dengan cukup baik. Pada subbab ini akan diberikan cara menghitung tingkat akurasi sistem dengan menggunakan Persamaan 6.1.

$$Akurasi = \frac{Hasil\ Benar}{Jumlah\ citra\ uji} \times 100\% \quad (6.1)$$

dengan Hasil Benar adalah jumlah citra wajah yang dikenali oleh sistem.

Perhitungan akurasi dilakukan pada data *training* dan data *testing*. Data *training* yang diuji terdiri dari 160 citra wajah individu, ditunjukkan pada Tabel 4. Sedangkan data *testing* yang diuji terdiri dari 80 citra wajah individu yang tidak bergerak, ditunjukkan pada Tabel 5 dan data *testing* dan 40 citra wajah individu yang bergerak oleh Tabel 6. Hasil uji coba secara lengkap dapat dilihat dalam LAMPIRAN 3. Berdasarkan hasil pengujian pada Tabel 4 sampai dengan Tabel 13 di dapatkan perbandingan akurasi sistem sebagai berikut,

Tabel 14. Akurasi Sistem secara Keseluruhan

Data Uji Coba	Uji 1	Uji 2	Uji 3	Uji 4	Uji 5
Data <i>Training</i>	98,13%	96,25%	93,13%	94,38%	61,88%
Data <i>Testing</i> citra wajah individu yang tidak bergerak	85%	85%	66,25%	62,5%	35%
Data <i>Testing</i> citra wajah individu yang bergerak	72,5%	50%	50%	45%	0%
Rata-Rata Akurasi	85,21%	77,08	69,79%	67,29%	32.29%

Dilihat dari hasil akurasi sistem bahwa untuk nilai akurasi pada sistem, ukuran penyamaan citra sangat berpengaruh terhadap hasil pada sistem. Secara keseluruhan akurasi tertinggi terdapat dari hasil uji coba 1 dengan ukuran citra 64x64 piksel dengan $T = 59$, rata-rata akurasi 85,21%. Sehingga untuk akurasi sistem dengan data *training* didapatkan hasil 98,13%, untuk akurasi sistem dengan seluruh data *testing* didapatkan hasil sebesar 80,8% dengan rincian 85% untuk akurasi citra wajah individu yang tidak bergerak dan 72,5% untuk akurasi citra wajah individu yang bergerak.

6.5 Pembahasan Penyebab Besar Kecil Akurasi

Hasil dari akurasi sistem menunjukkan perbedaan hasil antara hasil *testing* dengan citra wajah individu yang tidak bergerak dan hasil *testing* dengan citra wajah individu yang bergerak. Penyebab utama perbedaan akurasi ini adalah

sedikitnya data *training* citra wajah yang dilatih, sehingga ketika diujikan dengan data *testing* citra wajah individu bergerak yang notabene menangkap banyak ekspresi dan posisi dari individu, hasil dari tingkat akurasi pengenalan akan berkurang. Perbedaan ekspresi dan posisi dalam data yang di uji dapat dilihat dalam Gambar 21 sampai dengan Gambar 23.



Gambar 6.1. Data *Training* Afif



Gambar 6.2. Data *Testing* Afif (tidak bergerak)



Gambar 6.3.*Data Testing Afif (bergerak)*

BAB VII

KESIMPULAN DAN SARAN

7.1 Kesimpulan

Berdasarkan pembahasan hasil pengujian yang telah dilakukan terhadap sistem pengenalan individu melalui identifikasi wajah menggunakan metode *Hidden Markov Model* telah berhasil diaplikasikan dan dapat diambil beberapa kesimpulan sebagai berikut:

1. Pada penelitian ini telah berhasil dibuat sistem pengenalan individu melalui identifikasi wajah menggunakan metode *Hidden Markov Model*, dengan tahapan prosesnya yaitu *preprocessing*, ekstraksi fitur dengan *Singular Value Decomposition*, dan proses pengenalan dengan Metode *Hidden Markov Model*.
2. Hasil pengujian secara keseluruhan dari penelitian ini adalah akurasi tertinggi terdapat dari hasil uji coba 1 dengan ukuran gambar 64x64 piksel dengan $T = 59$, rata-rata akurasi 85,21%. Sehingga untuk akurasi sistem dengan data *training* didapatkan hasil 98,13%, untuk akurasi sistem dengan seluruh data *testing* didapatkan hasil sebesar 80,8% dengan rincian 85% untuk akurasi citra wajah individu yang tidak bergerak dan 72,5% untuk akurasi citra wajah individu yang bergerak.

7.2 Saran

Dengan melihat hasil yang dicapai pada penelitian ini, penulis menyarankan untuk menambah keberagaman data *training* untuk citra wajah individu dan percobaan

menggunakan ekstraksi fitur yang lain untuk mendapatkan fitur yang merepresentasikan citra secara akurat.

























DAFTAR PUSTAKA

















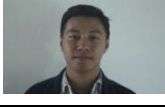











- [1] Prabhakar, S. , Pankanti, S. , and Jain, A (2003). “*Biometric Recognition: Security and Privacy Concerns*”, Proc IEEE Computer Society, IEE Press, p. 33.
- [2] Prasetyo, M. Eko Budi (2010). “Teori Dasar Hidden Markov Model”. Sekolah Teknik Elektro dan Informatika Institut Teknologi Bandung, Bandung.
- [3] Zufar, M.(2016).” *Convolutional Neural Networks* untuk Pengenalan Wajah Secara *Real-Time*” , Jurusan Matematika – FMIPA, Institut Teknologi Sepuluh Nopember, Surabaya.
- [4] Pandiangan, Harmuda, (2016). “Prediksi Sekuens DNA Berdasarkan Data Ekspresi Gen Menggunakan Model Hidden Markov”. Tesis, Jurusan Matematika – FMIPA, Institut Teknologi Sepuluh Nopember, Surabaya.
- [5] Sepritahara, (2012). “Sistem Pengenalan Wajah (*Face Recognition*) Menggunakan Hidden Markov Model (HMM)”, Jurusan Teknik Elektro Universitas Indonesia, Depok.
- [6] Elmezain, Mahmud dkk (2009). “*A Hidden Markov Model-Based Isolated and Meaningful Hand Gesture Recognition*”, *International Journal of Electrical, Computer, and Systems Engineering* 3:3 2009
- [7] Gonzalez, R.C and Rafael E.W(2008). “*Digital Image Processing*”. Prentice-Hall. Inc. United State, America.
- [8] Sutoyo. T. et al. (2009). “Teori Pengolahan Citra Digital”, Penerbit ANDI, Yogyakarta.





























- [9] Corcoran, P. M. dan Iancu, C(2011).” *Automatic Face Recognition System for Hidden Markov Model Techniques*”, Published by INTECH. College of Engineering & Informatics, National University of Ireland Galway, Ireland.
- [10] Fatta, Hanif Al (2007) “Konversi Format Citra RGB ke Format *Grayscale* Menggunakan Visual Basic’, Yogyakarta
- [11] Dinkova, Petya dkk. (2015). “*Face Recognition using Singular Value Decomposition and Hidden Markov Models*”. Mathematical Methods in Science and Mechanics
- [12] H.Miar-Naimi P.Davari.(2008) ”A New Fast And efficient HMM based Face Recognition System using a 7 state HMM along with SVD coefficient.”, International Journal of Electronics and Electrical Engg Vol 4.
- [13] H. Demirel, G. Anbarjafari, and M. N. S. Jahromi.(2008) “*Image equalization based on singular value decomposition*”, 23rd IEEE International Symposium on Computer and Information Sciences, pp. 1-5, Turkey.
- [14] G. Strang.(1993) “*Introduction to Linear Algebra.*” Wellesley-Cambridge Press, England.
- [15] Viola, P., Jones, M. J.(2001) “*Rapid Object Detection Using A Boosted Cascade of Simple Features*”, IEEE Conference on Computer Vision and Pattern Recognition, Jauai, Hawaii.
- [16] Rabiner L. R.(1989) “*A Tutorial Hidden Markov Models and Selected Application in Speech Recognition,*” Proceeding Of the IEEE, Vol. 77, No. 2, pp. 257-258.





























LAMPIRAN 1





























Citra Wajah *Training*





















Nama	Posisi			
	Depan	Kanan	Kiri	Atas
Adit				
Afif				
Agus				
Alfin				
Amel				
Anas				

Nama	Depan	Kanan	Kiri	Atas
Anin				
Anshar				
Ardi				
Ardi F				
Danni				
Aufa				
Ayu				

Nama	Depan	Kanan	Kiri	Atas
Bhara				
Brian				
Corry				
Dasilva				
Dian				
Dita				
Fadhlan				

Nama	Depan	Kanan	Kiri	Atas
Faizin				
Feri				
Gery				
Hayu				
Indra				
Intan				
Isabela				

Nama	Depan	Kanan	Kiri	Atas
Ivan				
Khoirul				
Lingga				
Nadya				
Nirwana				
Nuke				
Romli				

Nama	Depan	Kanan	Kiri	Atas
Rozi				
Satria				
Syifaul				
Uyun				
Wawan				

LAMPIRAN 2

Source Code

```
eps=.000001;
fprintf ('Loading Faces ...\n');
data_folder_contents = dir ('./Data wajah
training');
Database = cell(0,0);
person_index = 0;
max_coeffs = [-Inf -Inf -Inf];
min_coeffs = [ Inf  0  0];
for person=1:size(data_folder_contents,1);
    if
        (strcmp(data_folder_contents(person,1).name, '
.') || ...
        strcmp(data_folder_contents(person,1).name, '.
.') || ...
        (data_folder_contents(person,1).isdir
== 0))
        continue;
    end
    person_index = person_index+1;
    person_name =
data_folder_contents(person,1).name;
    Database{1,person_index} = person_name;
    fprintf([person_name, ' ']);
    person_folder_contents = dir(['./Data
wajah training/',person_name,'/*.jpg']);
    blk_cell = cell(0,0);
    for face_index=1:4
% --- Proses Preprocessing-----
I = imread(['./Data wajah
training/',person_name,'/',person_folder_cont
ents(face_index,1).name]);
% --- Proses Deteksi wajah-----
faceDetector = vision.CascadeObjectDetector;
bboxes = step(faceDetector,I);
handles.IFaces = insertObjectAnnotation(I,
'rectangle', bboxes, 'Face');
```

```

% --- Proses Cropping-----
I3 = imcrop(I,bboxes);
a=imresize(I3,[64,64]);
Im = rgb2gray(a); %Proses grayscale
or = ordfilt2(Im,1,true(3)); %Proses
filtering

% --- Proses Ekstraksi fitur-----
blk_index = 0;
for blk_begin=1:59
    blk_index=blk_index+1;
    blk = Im(blk_begin:blk_begin+5,:);
    [U,S,V] = svd(double(blk)); %Proses SVD
    blk_coeffs = [U(1,1) S(1,1) S(2,2)];
    %pengambilan koefisien SVD
    max_coeffs = max([max_coeffs;blk_coeffs]);
    min_coeffs = min([min_coeffs;blk_coeffs]);
    blk_cell{blk_index,face_index} = blk_coeffs;
        end
    end
    Database{2,person_index} = blk_cell;
    if (mod(person_index,10)==0)
        fprintf('\n');
    end
end
delta = (max_coeffs-min_coeffs)./([18 10 7]-
eps);
minmax = [min_coeffs;max_coeffs;delta];
for person_index=1:40
    for image_index=1:4
        for block_index=1:59%ekstraksi blok
            blk_coeffs = Database{2,person_index}
            {block_index,image_index};
            min_coeffs = minmax(1,:);
            delta_coeffs = minmax(3,:);
            qt = floor((blk_coeffs-min_coeffs)./
            delta_coeffs); %kuantisasi vektor
            Database{3,person_index}{block_index,image_in
            dex} = qt;
            label = qt(1)*10*7+qt(2)*7+qt(3)+1;%Pelabelan

```

```

Database{4,person_index}{block_index,image_index} = label;
end
Database{5,person_index}{1,image_index} =
cell2mat(Database{4,person_index}(:,image_index));
end
end
% --- Proses Training-----
transmat1(5,5) = 1;
for r=1:4
    transmat1(r,r) = 0.5;
    transmat1(r,r+1) = 0.5;
end

obsmat1 = (1/1260)*ones(5,1260);
fprintf('\nTraining ...\n');
for person_index=1:40
    fprintf([Database{1,person_index}, ' ']);
    seqmat =
cell2mat(Database{5,person_index});
[ESTTR,ESTEMIT]=hmmtrain(seqmat,transmat1,obs
mat1,'Tolerance',.01,'Maxiterations',20,'Algo
rithm','BaumWelch');
    mESTTR = max(ESTTR,eps);
    mESTEMIT = max(ESTEMIT,eps);
    Database{6,person_index}{1,1} = mESTTR;
    Database{6,person_index}{1,2} = mESTEMIT;
    if (mod(person_index,10)==0);
        fprintf('\n');
    end
end
fprintf('done.\n');
save DATA64 Database minmax

% --- Proses Testing-----
load('DATA64.mat');
min_coeffs = minmax(1,:);
max_coeffs = minmax(2,:);
delta_coeffs = minmax(3,:);

```

```

seq = zeros(1,59);
for blk_begin=1:59
blk = handles.or(blk_begin:blk_begin+5,:);
[U,S,V] = svd(double(blk));
blk_coeffs = [U(1,1) S(1,1) S(2,2)];
blk_coeffs = max([blk_coeffs;min_coeffs]);
blk_coeffs = min([blk_coeffs;max_coeffs]);
qt = floor((blk_coeffs-
min_coeffs)./delta_coeffs);
label = qt(1)*7*10+qt(2)*7+qt(3)+1;
seq(1,blk_begin) = label;
end

% --- Proses perhtiungan probabilitas-----
number_of_persons_in_database =
size(Database,2);
results =
zeros(1,number_of_persons_in_database);
for l=1:number_of_persons_in_database
TRANS = Database{6,1}{1,1};
EMIS = Database{6,1}{1,2};
PRIOR = [0.2 0.2 0.2 0.2 0.2];
n=length(TRANS(1,:));
T=length(seq);
for i=1:n           %initilization
    m(1,i)=EMIS(i,seq(1))*PRIOR(i);
end
for t=1:(T-1)       %recurssion
    for j=1:n
        z=0;
        for i=1:n
            z=z+TRANS(i,j)*m(t,i);
        end
        m(t+1,j)=z*EMIS(j,seq(t+1));
    end
end
p=0;
for i=1:n           %termination
    p=p+m(T,i);
end
end

```

```

        results(1,1) = p;
end

% --- Proses pencarian likelihood tertinggi-
-----
[maksimum, person_index] = max(results);
nama= Database{1,person_index};
fprintf(['This person is ',nama, '\n']);
        set(handles.text5, 'String', nama);

% --- Proses pengambilan video-----
function pushbutton1(~,~)
    [filename,pathname] =
    uigetfile('*.');
    if ~isequal(filename,0);
        vid =
        fullfile(pathname,filename);
        set(edit1, 'String', filename)
        set(btn2, 'Enable', 'on')
    else
        return;
    end
end

% --- tombol untuk memutar video-----
function pushbutton2(~,~)
    obj.reader =
    vision.VideoFileReader(vid);
    while ~isDone(obj.reader)
        [frame,frameSize] = readFrame();
        frame = im2uint8(frame);
        showFrameOnAxis(ax1,frame)
    end
    release(obj.reader);

    function [frame,frameSize] =
    readFrame()
        frame = obj.reader.step();
        R = frame(:, :, 1);

```

```

        G = frame(:,:,2);
        B = frame(:,:,3);
        frame = obj.reader.step();
        frameSize = size(frame);
    end
    function showFrameOnAxis(hAxis,~)
        try
            hChild = get(hAxis,
'Children');
            catch %#ok<CTCH>
                return; % hAxis does not
exist; nothing to draw
            end
            isFirstTime = isempty(hChild);
            if isFirstTime
                hIm = displayImage(hAxis,
frame);
                addScrollPanel(hAxis, hIm);
            else
                hIm = hChild;
                try
                    set(hIm,'cdata',frame);
drawnow;
                catch %#ok<CTCH>
                    % figure closed
                    return;
                end
            end
        end

        function addScrollPanel(hAxis, hIm)
            hPanel = get(hAxis,'parent');
            pos = get(hAxis,'position');

            hSP = imscrollpanel(hPanel,hIm);
            set(hSP,'Units','normalized',
'Position',pos);
        end
    end
    %-----
    -----

```

```

function hIm = displayImage(hAxis, ~)
    xdata = [1 frameSize(2)];
    ydata = [1 frameSize(1)];
    cdata = frame;
    cdatamapping = 'direct';
    hIm = image(cdata, ...
        'BusyAction', 'cancel', ...
        'Parent', hAxis, ...
        'CDataMapping', cdatamapping,
    ...
        'Interruptible', 'off');

    set(hAxis, ...
        'YDir', 'reverse', ...
        'TickDir', 'out', ...
        'XGrid', 'off', ...
        'YGrid', 'off', ...
        'PlotBoxAspectRatioMode',
    'auto', ...
        'Visible', 'off');
    end
end

% --- Proses ekstraksi video menjadi foto----
function pushbutton3(~,~)

    videoObject = VideoReader(vid)
    numberOfFrames =
    videoObject.NumberOfFrames;
    numberOfFramesWritten = 0;
    [folder, baseFileName, extentions] =
    fileparts(vid);
    folder = pwd;
    outputFolder = sprintf('%s/Movie
Frames from %s', folder, baseFileName);
    if ~exist(outputFolder, 'dir')
    mkdir(outputFolder);
    end
    numFrames = videoObject.NumberOfFrames;

```

```

numFramesWritten = 0;
for t = 1 : numFrames
    currFrame = read(videoObject, t);    %reading
    individual frames
    opBaseFileName = sprintf('%3.3d.png', t);
    opFullFileName = fullfile(outputFolder,
    opBaseFileName);
    imwrite(currFrame, opFullFileName, 'png');
    progIndication = sprintf('Wrote frame %4d of
    %d.', t, numFrames);
    disp(progIndication);
    numFramesWritten = numFramesWritten + 1;
end
progIndication = sprintf('Wrote %d frames to
folder "%s"', numFramesWritten, outputFolder);
disp(progIndication);
msgbox('Finish Extract Frame');
end

```


LAMPIRAN 3 HASIL UJI COBA

A. Hasil Uji Coba 1

Uji Coba Data Training			
No.	Nama	Dikenali dengan benar	Dikenali dengan salah
1	Adit	4	0
2	Afif	4	0
3	Agus	4	0
4	Alfin	4	0
5	Amel	4	0
6	Anas	4	0
7	Anin	4	0
8	Anshar	4	0
9	Ardi	3	1
10	Ardif	4	0
11	Aufa	3	1
12	Ayu	4	0
13	Bhara	3	1
14	Bryan	4	0
15	Corry	4	0
16	Dani	4	0
17	Dasilva	4	0
18	Dian	4	0
19	Dita	4	0
20	Fadhlan	4	0
21	Faizin	4	0
22	Fella	4	0
23	Feri	4	0
24	Gery	4	0
25	Hayu	4	0

26	Indra	4	0
27	Intan	4	0
28	Isabella	4	0
29	Ivan	4	0
30	Khoirul	4	0
31	Lingga	4	0
32	Nadya	4	0
33	Nirwana	4	0
34	Nuke	4	0
35	Romli	4	0
36	Rozi	4	0
37	Satria	4	0
38	Sifaul	4	0
39	Uyun	4	0
40	Wawan	4	0
	Akurasi	98,125%	1,875%

*Uji coba dilakukan pada 40 individu dengan masing-masing individu terdapat 4 citra wajah.

Uji Coba Data Testing 80 Gambar Individu yang Tidak Bergerak			
No.	Nama	Dikenali dengan benar	Dikenali dengan salah
1	Adit	2	0
2	Afif	2	0
3	Agus	2	0
4	Alfin	2	0
5	Amel	2	0
6	Anas	2	0
7	Anin	2	0
8	Anshar	2	0
9	Ardi	2	0
10	Ardif	1	1
11	Aufa	2	0

12	Ayu	2	0
13	Bhara	2	0
14	Bryan	2	0
15	Corry	2	0
16	Dani	0	2
17	Dasilva	2	0
18	Dian	2	0
19	Dita	2	0
20	Fadhlan	2	0
21	Faizin	2	0
22	Fella	2	0
23	Feri	0	2
24	Gery	1	1
25	Hayu	2	0
26	Indra	1	1
27	Intan	2	0
28	Isabella	2	0
29	Ivan	2	0
30	Khoirul	1	1
31	Lingga	2	0
32	Nadya	1	1
33	Nirwana	2	0
34	Nuke	1	1
35	Romli	1	1
36	Rozi	2	0
37	Satria	2	0
38	Sifaul	1	1
39	Uyun	2	0
40	Wawan	2	0
	Akurasi	85%	15%

*Uji coba dilakukan pada 40 individu dengan masing-masing individu terdapat 2 citra wajah.

Uji Coba Data Testing 40 Gambar Individu yang Bergerak				
No	Nama	Dikenali dengan benar	Dikenali dengan salah	Akurasi
1	Afif	11	9	55%
2	Dasilva	18	2	90%
Total Akurasi				72,5%

*Uji coba dilakukan pada 40 frame video

B. Hasil Uji Coba 2

Uji Coba Data Training			
No.	Nama	Dikenali dengan benar	Dikenali dengan salah
1	Adit	4	0
2	Afif	4	0
3	Agus	4	0
4	Alfin	4	0
5	Amel	4	0
6	Anas	4	0
7	Anin	4	0
8	Anshar	4	0
9	Ardi	4	0
10	Ardif	4	0
11	Aufa	3	1
12	Ayu	4	0
13	Bhara	4	0
14	Bryan	4	0
15	Corry	3	1
16	Dani	3	1
17	Dasilva	4	0
18	Dian	4	0
19	Dita	4	0

20	Fadhlan	3	1
21	Faizin	4	0
22	Fella	4	0
23	Feri	4	0
24	Gery	4	0
25	Hayu	4	0
26	Indra	4	0
27	Intan	4	0
28	Isabella	4	0
29	Ivan	4	0
30	Khoirul	4	0
31	Lingga	4	0
32	Nadya	4	0
33	Nirwana	4	0
34	Nuke	3	1
35	Romli	4	0
36	Rozi	3	1
37	Satria	4	0
38	Sifaul	4	0
39	Uyun	4	0
40	Wawan	4	0
	Akurasi	96,25%	3,75%

*Uji coba dilakukan pada 40 individu dengan masing-masing individu terdapat 4 citra wajah.

Uji Coba Data Testing 80 Gambar Individu yang Tidak Bergerak			
No.	Nama	Dikenali dengan benar	Dikenali dengan salah
1	Adit	2	0
2	Afif	2	0
3	Agus	2	0
4	Alfin	2	0

5	Amel	2	0
6	Anas	2	0
7	Anin	2	0
8	Anshar	2	0
9	Ardi	2	0
10	Ardif	1	1
11	Aufa	2	0
12	Ayu	2	0
13	Bhara	2	0
14	Bryan	1	1
15	Corry	2	0
16	Dani	1	1
17	Dasilva	2	0
18	Dian	1	1
19	Dita	2	0
20	Fadhlan	1	1
21	Faizin	2	0
22	Fella	2	0
23	Feri	1	1
24	Gery	2	0
25	Hayu	2	0
26	Indra	1	1
27	Intan	2	0
28	Isabella	2	0
29	Ivan	1	1
30	Khoirul	1	1
31	Lingga	2	0
32	Nadya	1	1
33	Nirwana	2	0
34	Nuke	1	1
35	Romli	2	0
36	Rozi	2	0
37	Satria	2	0
38	Sifaul	1	1

39	Uyun	2	0
40	Wawan	2	0
	Akurasi	85%	15%

*Uji coba dilakukan pada 40 individu dengan masing-masing individu terdapat 2 citra wajah.

Uji Coba Data Testing 40 Gambar Individu yang Bergerak				
No	Nama	Dikenali dengan benar	Dikenali dengan salah	Akurasi
1	Afif	0	20	0%
2	Dasilva	20	0	100%
Total Akurasi				50%

*Uji coba dilakukan pada 40 frame video

C. Hasil Uji Coba 3

Uji Coba Data Training			
No.	Nama	Dikenali dengan benar	Dikenali dengan salah
1	Adit	4	0
2	Afif	4	0
3	Agus	4	0
4	Alfin	4	0
5	Amel	4	0
6	Anas	4	0
7	Anin	4	0
8	Anshar	4	0
9	Ardi	3	1
10	Ardif	4	0
11	Aufa	3	1
12	Ayu	4	0

13	Bhara	4	0
14	Bryan	3	1
15	Corry	3	1
16	Dani	3	1
17	Dasilva	4	0
18	Dian	3	1
19	Dita	4	0
20	Fadhlan	4	0
21	Faizin	3	1
22	Fella	4	0
23	Feri	4	0
24	Gery	4	0
25	Hayu	3	1
26	Indra	4	0
27	Intan	4	0
28	Isabella	4	0
29	Ivan	4	0
30	Khoirul	4	0
31	Lingga	4	0
32	Nadya	4	0
33	Nirwana	3	1
34	Nuke	3	1
35	Romli	4	0
36	Rozi	3	1
37	Satria	4	0
38	Sifaul	4	0
39	Uyun	4	0
40	Wawan	4	0
	Akurasi	93,125%	6,875%

*Uji coba dilakukan pada 40 individu dengan masing-masing individu terdapat 4 citra wajah.

Uji Coba Data Testing 80 Gambar Individu yang Tidak Bergerak			
No.	Nama	Dikenali dengan benar	Dikenali dengan salah
1	Adit	2	0
2	Afif	2	0
3	Agus	1	1
4	Alfin	1	1
5	Amel	0	2
6	Anas	0	2
7	Anin	2	0
8	Anshar	2	0
9	Ardi	2	0
10	Ardif	1	1
11	Aufa	2	0
12	Ayu	1	1
13	Bhara	0	2
14	Bryan	2	0
15	Corry	1	1
16	Dani	1	1
17	Dasilva	2	0
18	Dian	1	1
19	Dita	1	1
20	Fadhlan	1	1
21	Faizin	2	0
22	Fella	2	0
23	Feri	0	2
24	Gery	1	1
25	Hayu	1	1
26	Indra	1	1
27	Intan	2	0
28	Isabella	2	0
29	Ivan	0	2
30	Khoirul	1	1

31	Lingga	2	0
32	Nadya	1	1
33	Nirwana	2	0
34	Nuke	0	2
35	Romli	2	0
36	Rozi	2	0
37	Satria	2	0
38	Sifaul	1	1
39	Uyun	2	0
40	Wawan	2	0
	Akurasi	66,25%	33,75%

*Uji coba dilakukan pada 40 individu dengan masing-masing individu terdapat 2 citra wajah.

Uji Coba Data Testing 40 Gambar Individu yang Bergerak				
No	Nama	Dikenali dengan benar	Dikenali dengan salah	Akurasi
1	Afif	0	20	0%
2	Dasilva	20	0	100%
Total Akurasi				50%

*Uji coba dilakukan pada 40 frame video

D. Hasil Uji Coba 4

Uji Coba Data Training			
No.	Nama	Dikenali dengan benar	Dikenali dengan salah
1	Adit	4	0
2	Afif	4	0
3	Agus	4	0
4	Alfin	4	0

5	Amel	4	0
6	Anas	4	0
7	Anin	4	0
8	Anshar	4	0
9	Ardi	3	1
10	Ardif	4	0
11	Aufa	2	1
12	Ayu	4	0
13	Bhara	4	0
14	Bryan	3	1
15	Corry	4	0
16	Dani	3	1
17	Dasilva	4	0
18	Dian	4	0
19	Dita	4	0
20	Fadhlan	4	0
21	Faizin	3	1
22	Fella	4	0
23	Feri	4	0
24	Gery	4	0
25	Hayu	3	1
26	Indra	4	0
27	Intan	4	0
28	Isabella	4	0
29	Ivan	4	0
30	Khoirul	3	1
31	Lingga	4	0
32	Nadya	4	0
33	Nirwana	3	1
34	Nuke	4	0
35	Romli	4	0
36	Rozi	4	0
37	Satria	4	0
38	Sifaul	4	0

39	Uyun	4	0
40	Wawan	4	0
	Akurasi	94,375%	5,625%

*Uji coba dilakukan pada 40 individu dengan masing-masing individu terdapat 4 citra wajah.

Uji Coba Data Testing 80 Gambar Individu yang Tidak Bergerak			
No.	Nama	Dikenali dengan benar	Dikenali dengan salah
1	Adit	2	0
2	Afif	2	0
3	Agus	2	0
4	Alfin	2	0
5	Amel	0	2
6	Anas	0	2
7	Anin	1	1
8	Anshar	2	0
9	Ardi	2	0
10	Ardif	1	1
11	Aufa	2	0
12	Ayu	1	1
13	Bhara	0	2
14	Bryan	2	0
15	Corry	1	1
16	Dani	1	1
17	Dasilva	2	0
18	Dian	1	1
19	Dita	1	1
20	Fadhlan	1	1
21	Faizin	1	1
22	Fella	2	0
23	Feri	0	2
24	Gery	0	2

25	Hayu	0	2
26	Indra	1	1
27	Intan	2	0
28	Isabella	2	0
29	Ivan	0	2
30	Khoirul	1	1
31	Lingga	2	0
32	Nadya	1	1
33	Nirwana	2	0
34	Nuke	0	2
35	Romli	2	0
36	Rozi	1	1
37	Satria	2	0
38	Sifaul	1	1
39	Uyun	2	0
40	Wawan	2	0
	Akurasi	62,5%	37,5%

*Uji coba dilakukan pada 40 individu dengan masing-masing individu terdapat 2 citra wajah.

Uji Coba Data Testing 40 Gambar Individu yang Bergerak				
No	Nama	Dikenali dengan benar	Dikenali dengan salah	Akurasi
1	Afif	0	20	0%
2	Dasilva	18	2	90%
Total Akurasi				45%

*Uji coba dilakukan pada 40 frame video

E. Hasil Uji Coba 5

Uji Coba Data Training			
No.	Nama	Dikenali dengan benar	Dikenali dengan salah
1	Adit	3	1
2	Afif	4	0
3	Agus	4	0
4	Alfin	1	3
5	Amel	3	1
6	Anas	3	1
7	Anin	2	2
8	Anshar	1	3
9	Ardi	2	2
10	Ardif	3	1
11	Aufa	1	3
12	Ayu	2	2
13	Bhara	4	0
14	Bryan	3	1
15	Corry	2	2
16	Dani	2	2
17	Dasilva	0	4
18	Dian	2	2
19	Dita	4	0
20	Fadhlan	3	1
21	Faizin	1	3
22	Fella	0	4
23	Feri	4	0
24	Gery	4	0
25	Hayu	1	3
26	Indra	3	1
27	Intan	0	4
28	Isabella	4	0
29	Ivan	3	1

30	Khoirul	0	4
31	Lingga	3	1
32	Nadya	3	1
33	Nirwana	3	1
34	Nuke	4	0
35	Romli	3	1
36	Rozi	2	2
37	Satria	4	0
38	Sifaul	3	1
39	Uyun	2	2
40	Wawan	3	1
	Akurasi	61,875%	38,125%

*Uji coba dilakukan pada 40 individu dengan masing-masing individu terdapat 4 citra wajah.

Uji Coba Data Testing 80 Gambar Individu yang Tidak Bergerak			
No.	Nama	Dikenali dengan benar	Dikenali dengan salah
1	Adit	1	1
2	Afif	1	1
3	Agus	2	0
4	Alfin	0	2
5	Amel	0	2
6	Anas	0	2
7	Anin	0	2
8	Anshar	2	0
9	Ardi	1	1
10	Ardif	1	1
11	Aufa	0	2
12	Ayu	0	2
13	Bhara	0	2
14	Bryan	1	1

15	Corry	1	1
16	Dani	1	1
17	Dasilva	0	2
18	Dian	1	1
19	Dita	0	2
20	Fadhlan	0	2
21	Faizin	0	2
22	Fella	0	2
23	Feri	0	2
24	Gery	2	0
25	Hayu	0	2
26	Indra	0	2
27	Intan	0	2
28	Isabella	0	2
29	Ivan	0	2
30	Khoirul	0	2
31	Lingga	1	1
32	Nadya	1	1
33	Nirwana	2	0
34	Nuke	1	1
35	Romli	1	1
36	Rozi	0	2
37	Satria	2	0
38	Sifaul	2	0
39	Uyun	2	0
40	Wawan	2	0
	Akurasi	35%	65%

*Uji coba dilakukan pada 40 individu dengan masing-masing individu terdapat 2 citra wajah.

Uji Coba Data Testing 40 Gambar Individu yang Bergerak				
No	Nama	Dikenali dengan benar	Dikenali dengan salah	Akurasi
1	Afif	0	20	0%
2	Dasilva	0	20	0%
Total Akurasi				0%

*Uji coba dilakukan pada 40 frame video

BIODATA PENULIS



Nama lengkap penulis yaitu Alvina Khairun Nisa' yang biasa dipanggil Vina, lahir di Tulungagung, 25 Maret 1995. Pendidikan formal yang pernah ditempuh yaitu TK Dharma Wanita Persatuan Punjul Tulungagung pada tahun 2000-2001, SDI Al-Azhaar Tulungagung pada tahun 2001-2007, Mts Negeri Denanyar Jombang pada tahun 2007-2010, SMA Darul Ulum 2 Jombang pada tahun 2010-2013. Saat ini penulis sedang menempuh pendidikan S1 di Departemen Matematika Institut Sepuluh Nopember dengan bidang minat Ilmu Komputer. Selama di bangku kuliah, penulis aktif di organisasi dalam kampus yaitu HIMATIKA ITS. Pada tahun 2014-2015 penulis menjadi staff Departemen Hubungan Luar (HUBLU) HIMATIKA ITS. Pada tahun 2015-2016 penulis menjadi Kepala Biro Media Informasi Departemen External Affair (EXA) HIMATIKA ITS. Selain aktif dalam berorganisasi, penulis juga aktif dalam beberapa acara kepanitiaan, seperti menjadi OC Padamu HIMATIKA ITS pada tahun 2014, *Art and Decoration staff* dalam OMITS HIMATIKA ITS tahun 2015 dan OMITS HIMATIKA ITS tahun 2016. Demikian biodata penulis. Jika ingin memberi kritik, saran, dan diskusi mengenai Tugas Akhir ini, dapat dikirimkan melalui email alvina.khariun@gmail.com. Terima kasih.

